

Scan: S. Höltgen
Das Magazin für aktives und kreatives Computern
1Z 3409 E
DM 5,–
HC
9 September 1986
Mein Home-Computer
HC-EXTRA PASCAL
Große Schnittstellen-Übersicht
● Für Atari ● Commodore ● IBM PC
● Schneider ● Sinclair
PROLOG
Der Weg zur künstlichen Intelligenz
Die Sprache Prolog
Amstrad-Show in London
Neues für Schneider Fans
HC-Test
Farbdrucker im Vergleich
Aktiv Computern
C64: Neues Speed-DOS
Atari ST: Printer Spooler
Schneider CPC: Reversi
OKIMATE



Myrtle ist verliebt.
Immer wenn sie Gordon sieht, schlägt ihr Herz höher und sie versucht ihn mit der zarten Stimme eines Nebelhorns zu betören – worauf Gordon ins Wasser flüchtet.

Doch zu früh gefreut – Myrtle stürzt sich mit ihren 2 Zentnern in die Fluten und verwandelt sich in eine reizende – wenn auch etwas dickliche – Meerjungfrau.

In der phantastischen Unterwasserwelt sucht Myrtle nun verzweifelt nach Gordon. Versteckt er sich vielleicht hinter dem Wrack des riesigen Ozeandampfers, oder in der versunkenen Stadt oder ... Doch erst muß Myrtle ein Guiness trinken, um ihre verliebte Suche fortzusetzen.

Erhältlich für
Commodore 64 Cassette und Diskette
Schneider CPC Cassette und Diskette
ZX Spectrum 48K

Activision Deutschland GmbH,
Postfach 76 06 80, 2000 Hamburg 76.

VERTRIEB DEUTSCHLAND:
Ariolasoft (Exclusiv-Distributor)
Rushware (Autorisierter Mitvertrieb)
VERTRIEB ÖSTERREICH:
Karasoft (Exclusiv-Distributor)
VERTRIEB SCHWEIZ:
HILCU (Exclusiv-Distributor)

Vogel-Verlag
Redaktion HC
Schillerstraße 23a
8000 München 2

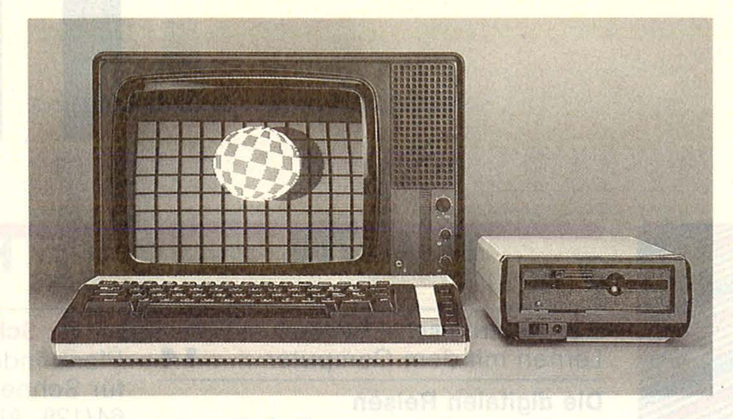
Für und wieder Pascal

Alles schön und gut, daß Sie sich jetzt so ausführlich mit Pascal beschäftigen – aber gibt es denn sonst gar nichts mehr? Ich würde mir wünschen, daß in Zukunft weiter mehr für Assembler-Programmierer geboten wird – schließlich ist Maschinensprache nicht nur schneller, sondern für spezielle Probleme auf der Betriebssystemebene immer noch die optimale Lösung.
HELMUT REK
4156 WILLICH

Warum entschuldigen sich Autoren, die Einführungen in Pascal bringen, eigentlich immer dafür, daß Pascal „umständlich" ist? Es ist doch im Gegenteil so, daß durch Vorgabe von globalen und lokalen Variablen bestimmter Typen von Anfang an Ordnung ins Programm kommt.
PROF. DR. DR. H. C. G. SNATZKE
4630 BOCHUM

Nichts gegen Pascal – aber ich finde es unfair, daß jetzt gegen BASIC geschossen wird und alle BASIC-Programmierer als Chaoten abgestempelt werden. Auch in BASIC läßt sich sauber programmieren, und wenn man das Endresultat dann durch einen Compiler jagt, sind die Programme nicht langsamer als die der Pascal-Konkurrenz.
RAINER WALDHOFF
8000 MÜNCHEN

Daß Pascal die Zukunftssprache ist, sehe ich noch nicht. Ein Programmiersystem der Zukunft kann meiner Meinung nach nur im Dialogsystem funktionieren, aber dafür reicht wohl der Speicherplatz der heutigen Home-Computer noch nicht aus.
KARL BAUMANN
6443 SONTRA



Amigas Nachfolger

HC 7/86, Seite 116
Der letzte Hammer
Als wir das Programm mit dem „Amiga-Ball" das erste Mal laufenließen, blieb auch uns die Spucke weg. Wir waren spätestens von da an überzeugt, daß unser Computer kein popeliger Atari 800 XL ist. Wir sind empört darüber, daß ein Computer, der zu solchen Leistungen fähig ist, so herabgewürdigt wird.
ATARI-USER-CLUB
2190 CUXHAVEN 1

In Ihrer Juli-Ausgabe haben Sie einen weiteren Beweis dafür geliefert, daß Sie von Commodore bestochen sind. Oder hätten Sie sonst den Preis/Leistungs-Mörder (hoher Preis/geringe Leistung) Commodore Amiga als Wundermaschine bezeichnet? Wenn Sie wirklich ein Programm vorweisen können, welches den Amiga-Ball nachmacht, so sei Ihnen vergeben. Im anderen Fall erwarte ich von Ihnen, daß Sie sich in aller Form bei allen Atarianern entschuldigen und in Zukunft solche perversen Veröffentlichungen unterlassen. Denn wenn man den Atari 800 XL neben den wirklich popeligen Commodore 64 hinstellt, so muß man doch zugeben, daß der Atari 800 XL ein weitaus schöneres Design vorweisen kann, als der Commodore 64.
RENÉ SCHUSTER
A-8020 GRAZ

Anmerkung der Redaktion: Das Programm, welches den Amiga-Ball aus der Werbung auf dem Atari 800 XL zum Laufen bringt, stammt aus den USA und liegt uns vor.

Unfähig!

Mir kommt es so vor, als könnten Sie einen Personal-Computer nicht von einem Home-Computer unterscheiden. Wie kann es sonst passieren, daß der Atari 260 ST in Ihrer Hitparade in der Kategorie „Home-Computer" landet. Daß ein Home-Computer einen 16-bit-Prozessor, einen Speicher von 512 KByte und eine GEM-Benutzeroberfläche besitzt, ist mir unbekannt. Mal sehen, wie lange es dauert, bis der Commodore 64 mit GEOS ein Personal-Computer geworden ist.
OLIVER BACZKOWSKI
1000 BERLIN 41

Zu viel für Commodore 64

Eigentlich sind fast alle Zeitschriften mit Programmen für C64 sehr reich gesegnet, so daß zum Beispiel ein „Sprite-Editor" wohl schon in allen Zeitschriften zu finden war und von da her nur Platz in Ihrer Zeitschrift verschwendet. Anstelle der sich immer wiederholenden Programme könnte man doch etwas für andere Computer bringen.
ANDREAS BORCHERT
2120 LÜNEBURG

Es lebe der C64 ...

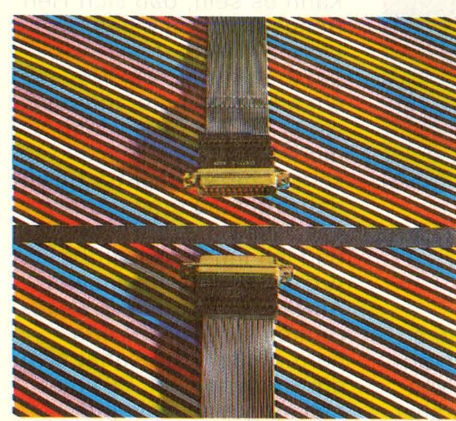
HC 7/86, Seite 3
Es lebe der Atari ...
Kann es sein, daß sich Herr Schuster, Autor eines Leserbriefs in Ihrer Juli-Ausgabe, erst seit zwei Wochen für Computer interessiert? Wenn er den Commodore 64 als Spielmaschine bezeichnet und sich dabei auf die große Anzahl von Spielen bezieht, so darf ich ihn an die große Anzahl an professionellen Anwendungen und Erweiterungen für den C64 erinnern. Außerdem eignet sich die Hardware des C64 nun mal so gut für Spielprogramme.

Der Commodore 64 wird wegen seinem schwachen BASIC neben dem Apple als Lern- und Ausbildungs-Computer eingesetzt. Das V.2-BASIC regt durch die wenigen Befehle zur eigenen Kreativität an und ist schnell und leicht zu erlernen und die hardwareorientierte Programmierung durch PEEK und POKE ist der erste Schritt zur Maschinensprache. Der heißen und lahmen 1541-Floppy kann man durch Speeder, Kühlventilatoren und einem stabilisierten Netzteil sehr schnell Abhilfe schaffen.
STEFAN OTTERSBACH
5207 RUPPICHTEROTH

Wie ist Ihre Meinung?

Wenn Ihnen zu einem Artikel in HC oder zur Meinung anderer Leser etwas einfällt, können Sie uns gerne schreiben. Wir freuen uns über Ihre Zuschriften. Je kürzer ein Leserbrief ist, um so größer ist seine Chance, veröffentlicht zu werden.
Die HC-Redaktion

INH



Kontakt zur Außenwelt: *Ohne Schnittstellen läuft beim Computer fast gar nichts. Ab Seite 18*



C16-User sind gefragt: *Wie bringt man die C64-Programme zum Laufen. Ab Seite 28*



Buntes Feuerwerk: *Die neuen Farbdrucker ab Seite 104*

MAGAZIN

WORKSHOP

HARDWARE

SOFTWARE

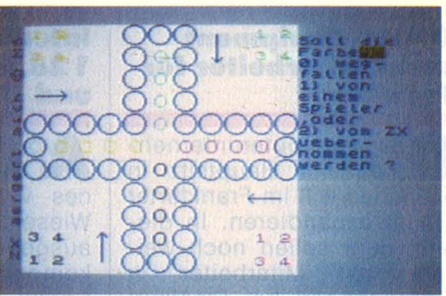
RUBRIKEN



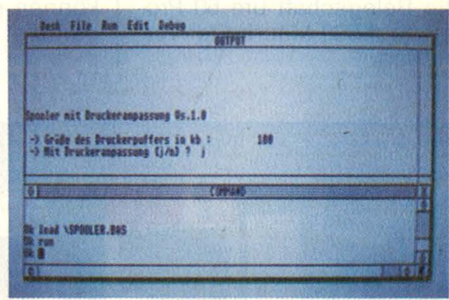
HC-EXTRA: PASCAL

AKTIV COMPUTERN

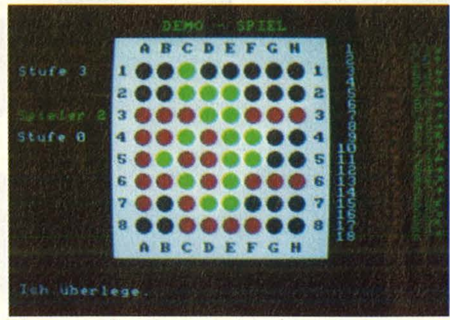
AKTIV COMPUTERN



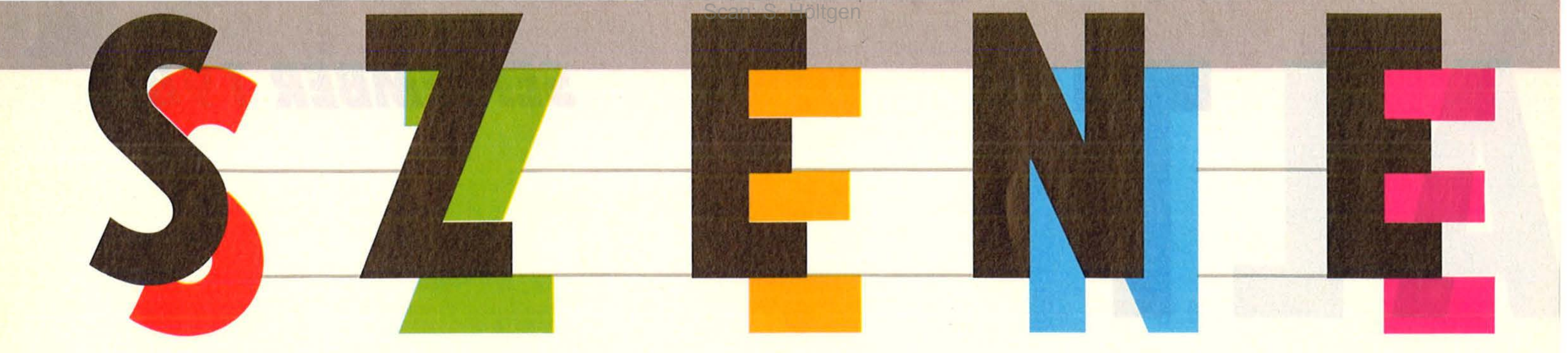
ZX-ärgert mich: *Gesellschaftsspiel für Spectrum ab Seite 72*



Printer Spooler *mit Druckeranpassung für Atari ST ab Seite 43*



Reversi: *Strategiespiel auf dem Schneider CPC ab Seite 35*

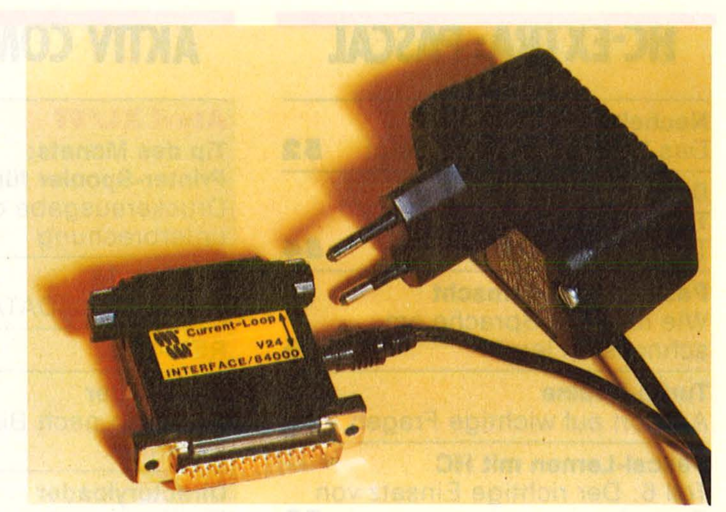


Digital Equipment sucht Mitarbeiter für Frankfurt

Das Computerunternehmen mit Hauptsitz in München will im Frankfurter Raum expandieren. In diesem Jahr sollen noch weitere 100 Mitarbeiter in Sprendlingen bei Frankfurt eingestellt werden. Bereits im vergangenen Jahr wurde die Belegschaft um 50 Prozent auf 500 Mitarbeiter vergrößert.

Interface für C64/128 jetzt Speeddos- und Geos-kompatibel

Ab sofort werden alle C64/128-Druckerinterfaces vom Typ 9200/G von Wiesemann in einer Version ausgeliefert, die vollständig kompatibel ist mit High-Speed Betriebssystemen wie Speeddos und Geos. Ältere Interfaces dieses Typs können gegen 50 Mark umgerüstet werden.





Current-Loop/V24-Interface

Das neue Interface Typ 84 000 von Wiesemann wandelt eine 20-mA-Schnittstelle in eine V24 um. Das Gerät überträgt bidirektional; Stromschleife und V24-Seite sind durch Optokoppler voneinander getrennt. Die Stromschnittstelle arbeitet passiv. Unterschiedliche Zuordnungen von Strom-/Nicht-Strom zu 0/1-Pegeln können im Interface angeglichen werden – jeder Kanal kann separat invertierend oder nicht-invertierend betrieben werden. Das Interface kann mit einem mitgelieferten Steckernetzteil oder über das angeschlossene V24-Gerät (5 Volt) versorgt werden. Das Interface ist für unter 250 Mark lieferbar.

Sorgt für Sicherheit und Ordnung

Aufstell-Ringbuchordner mit Ablegehüllen für Disketten bietet jetzt Döbbelin & Boeder. So sei eine platzsparende Archivierung von 40 Disketten im 8″-, 5,25″- oder 3,5″-Format möglich. Das Ringbuch ist aus blauem, festem Kunststoff und ist nicht nur stets griffbereit, sondern kann auch am Arbeitsplatz aufgestellt werden.

Was Commodore noch in der Schublade hat

Die Gerüchteküche ist am Brodeln. Im Herbst, so heißt es, soll endlich der Amiga 2000 in Deutschland vorgestellt werden. Also der mit eingebauter IBM PC-Kompatibilität und $5^1/_4$ Zollaufwerk. Damit man endlich vernünftig Textverarbeitung machen könne, werde gerade Vizawrite Classic für den Kreativ-Computer umgestrickt. Damit könnte die Maschine auch für Leute aus der schreibenden Zunft sehr interessant werden. Und noch ein Gerücht: Mit der Einführung des aufgemotzten Modells soll der Preis für den derzeitigen Amiga gesenkt werden, schließlich will man mit Atari mithalten können. Es wird sogar gemunkelt, es gäbe noch eine PAL-Version, damit die Einsteiger erst einmal am heimischen Farbfernseher die Qualitäten nutzen können.

Billigste PC

Die preiswerteste Quelle für IBM-Clones ist bekanntlich Taiwan. Und hier hält man sich auch nicht so eng an BIOS-Urheberrechte und so weiter und liefert gar mit Original-ROM aus. Wer billig an einen Kompatiblen kommen will, sollte einmal einen Vielflieger nach Bezugsmöglichkeiten fragen.

UNDERGROUND



Universeller Monitor für Home-Computer

Speziell auf die Bedürfnisse der Besitzer von Home-Computern zugeschnitten ist der Farbmonitor von Orion. Wegen des recht günstigen Preises von knapp 1000 Mark haben wir uns das Gerät genauer angesehen. Der Ergebnis: Flimmerfreie Farbgrafiken vom C64 präsentieren sich in kaum jemals erreichter Qualität.

Mit seinen vielfältigen Eingängen (RGG analog/digital, Composite, Luminanz/Crominanz) kann der C128 beispielsweise im RGB-Modus betrieben werden. Die Folge ist eine nochmals gesteigerte Brillanz, da bei dieser Betriebsart das Bildsignal dem Monitor in der Form zur Verfügung gestellt wird, mit der die Bildröhre direkt angesteuert werden kann. Jegliche (qualitätsmindernden) Umformungen entfallen hierbei.

In punkto Auflösung ließ der CCM-1280 keinen Wunsch offen: Auch kritische Farbkombinationen (etwa rote Schrift auf blauem Grund) wurden mühelos dargestellt. Angenehm fiel der Umschalter auf, der ein monochromes S/W-Bild in schwarzgrüne Darstellung verwandelt. Besonders bei einfarbigen Textprogrammen erwies sich diese Einstellung als ideal.

Eine weitere Anwendungsmöglichkeit ist die Verwendung als Sichtgerät für den heimischen Video-Recorder. Bei dieser Betriebsart erweist sich der Eingangswahlschalter als nützlich: Ein Umschalten von Computer- auf Videobild ist ohne Umstecken von Kabeln möglich. Daß dieser Schalter allerdings auf die Gehäuserückseite verbannt wurde, erscheint uns unverständlich. Als störend wurde auch empfunden, daß der vertikale Bildfang nach dem Umschalten (zum Glück auf der Frontseite) nachgeregelt werden mußte.

Thorsten Freiberg

DB25-Steckergehäuse auch für dünne Kabel

DB25-Stecker werden meist mit dünnen Kabeln mit drei bis zehn Adern eingesetzt. Übliche Steckergehäuse sind jedoch so konstruiert, daß die Zugentlastung für Kabel mit mindestens zehn Adern gedacht ist. Das Steckergehäuse GDB25 von Wiesemann, Wuppertal, hat dagegen eine Zugentlastung, die bereits Kabeldurchmesser ab zwei Millimeter fest packen kann. Großabnehmer (ab 1000 Stück) können gegen Aufpreis ihre spezielle Farbe geliefert bekommen. Stückpreis etwa 2 Mark.

TERMINE

Computertage an der Saar

Messe: Die 3. Elektronik- und Computertage Saar finden als Verkaufs- und Informationsmesse statt. Im Foyer West der Kongreßhalle ist das Home-Computer-Paradies zu finden mit allem Zubehör, was der Anwender braucht. Fachvorträge, unter anderem von der Fachhochschule und der Universität des Saarlandes, helfen bei Problemlösungen.
Saarbrücken, Kongreßhalle, 5.–7. 9. 86.

Foto- und Computerbörse

Flohmarkt: Die Fotogruppe '81 veranstaltet ihre 1. Aalener Foto- und Computerbörse. Gebrauchte Fotogeräte und Computer mit Zubehör und Peripherie dürfen angeboten und natürlich auch gekauft werden.
Aalen (Ostalbkreis), im evangelischen Gemeindehaus, 11–16 Uhr, 7. 9. 86. Info: 0 73 61/7 77 74.

Texten mit Joyce

Kurs: Locoscript, Textverarbeitung auf dem Joyce.
Berlin-Neukölln, jeden Dienstag und Donnerstag von 19–22 Uhr. Die Dauer des Kurses bestimmt der Lernende.
Veranst.: *„Ihre Computerei", Berlin 44. Info: 0 30/ 6 84 26 67.*

Software für Anwender

Kurse: Dbase III – Einführung.
24.–26. 9. 86.
F & A (Frage und Antwort).
25.–26. 9. 86.
Lotus 1-2-3, Vers. 2.0.
29.–30. 9. 86
Lotus Symphony.
22.–24. 9. 86.
Open Access II.
10.–12. 9. 86.
Alle Kurse in Hamburg.
Veranst.: Computer College. Info: 0 40/2 70 78.
Seminare: PC Oracle und 10-Net (kostenfrei).
Hannover, 12. 9. 86.
Berlin, 15. 9. 86.
Veranst.: Kettler EDV-Consulting. Info: 0 80 42/80 81.

Unix, 68000er und C

Kurse: Grundkurs Unix-Betriebssystem.
2.–4. 9. 86.
Programmierung unter Unix
9.–11. 9. 86.
Motorola 16-Bit MC 6800.
10.–11. 9. 86.
Einführung in die Programmiersprache C.
15.–19. 9. 86.
BASIC auf dem IBM PC.
17.–19. 9. 86.
Alle Kurse in Essen.
Veranst.: Haus der Technik e.V. Info: 02 01/18 03-1.

GSX-Literatur

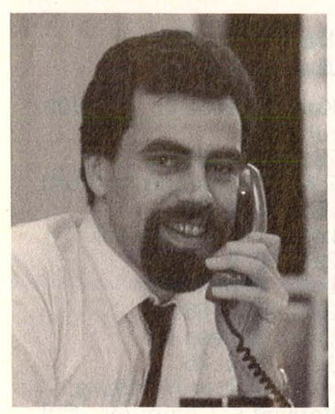
Die Schneiderdata GmbH in Freising hat den Vertrieb der Original-Handbücher zu GSX-80 der Digital Research Inc. übernommen. Die englische Literatur umfaßt einen GSX-80 User's Guide und einen GSX Programmers Guide. Das Gesamtpaket kostet um 150 Mark und ist bei Schneider-Vertragshändlern erhältlich.

Wo ist denn der Pharao geblieben?

Von Talent Computer Systems kommt das Spiel „The Lost Pharaoh" für den Sinclair QL. Tief unter dem heißen Wüstensand führt ein Labyrinth zu der Kammer des Pharaos. Schurkische Diebe und weniger schurkische Archäologen sind auf der Suche bereits verloren gegangen.

BLITZ INTERVIEW

Schneider: mit der Joyce auf dem Weg in die Zukunft



Fred Köster, Computer Division von Schneider

Im Juni hatten wir alle Beteiligten über den Verkauf der Vertriebs- und Produktionsrechte aller Sinclair-Computer an Amstrad befragt. Schneider gab damals bekannt, man plane, die Sinclair-Hardware im deutschsprachigen Raum zu vertreiben. Nähere Angaben fehlten damals allerdings. Wir fragten Fred Köster, Prokurist der Schneider AG und Leiter der Computer Division, nach dem neuesten Stand und sprachen auch mit ihm über die zukünftige Weiterentwicklung von Schneider.

Wann wird es die ersten Sinclair-Computer unter dem Schneider-Label geben?
Nach eingehenden Überlegungen haben wir uns entschlossen, die Sinclair-Produkte nicht zu vertreiben.

Worin liegt der Grund für diesen Umschwung?
Uns erscheint eine Integration dieser Computer in unsere bestehende Produktpalette von der Abgrenzung her nicht sinnvoll. Zum anderen haben wir uns entschieden, uns in Zukunft auf eine andere Richtung zu konzentrieren.

Welche denn?
Nun, wir haben ja mit Joyce den Weg bereits angedeutet.

Sie meinen den Bereich der Personal-Computer. Heißt das, daß der Schneider PC bereits auf der nächsten Computermesse, der Orgatechnik, zu sehen ist?
Dazu möchte ich zum jetzigen Zeitpunkt nichts sagen.

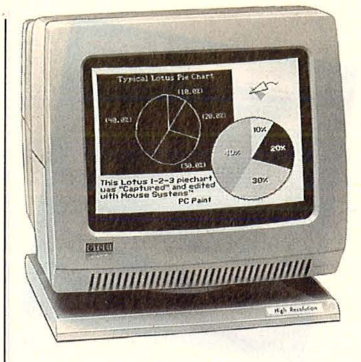
Glauben Sie, daß der Schneider PC gegenüber den preiswerten kompatiblen Taiwan-PC überhaupt noch eine Chance besitzt?
Wir werden sicherlich nur Produkte einführen, denen wir hervorragende Marktchancen einräumen. Um einen Preis bewerten zu können, muß man die Leistung und Ausstattung eines Computers zu diesem Preis in Beziehung bringen. Und da werden wir mit Sicherheit auch zukünftig die Nase vorn haben.

In letzter Zeit ist der Preis für den CPC6128 erheblich gefallen. War der Rechner zum alten Preis nicht mehr zu verkaufen?
Wir haben die Verkaufspreisempfehlungen gesenkt, da wir aufgrund günstiger Währungsentwicklungen sowie Rationalisierungen in der Produktion in der Lage dazu waren.

Wird es in der nächsten Zeit Neuheiten beziehungsweise Änderungen bei der CPC-Modellreihe geben?
Aktuell stehen keine Neuheiten bei dieser Produktfamilie vor ihrer unmittelbaren Markteinführung.

In unserem letzten Interview hat ihnen Winfried Hoffmann, der Geschäftsführer von Commodore Deutschland, Erfolglosigkeit vorgeworfen und Probleme prophezeit.
In dieser Branche wird anscheinend öfter einmal fahrlässig mit Marktzahlen umgesprungen – Schneider hat im Vergleich zum ersten Halbjahr 1985 in den Monaten Januar bis Juni 1986 ein Wachstum von 50% erreicht.
Von einem Abrutschen im Marktanteil kann also schon deshalb gar keine Rede sein, da der Gesamtmarkt deutlich langsamer gewachsen ist. Wir von Schneider wollen uns auch zukünftig darauf konzentrieren, unsere Produkte weiter erfolgreich zu vermarkten, anstelle negative Gerüchte über unseren Mitbewerb zu streuen.
Und außerdem werden wir für die Zukunft unserem soliden Vertriebskonzept treu bleiben. Wir wollen unter keinen Umständen unsere Computer über Lebensmittel-Discounter oder Tankstellen anbieten, da wir der Meinung sind, daß unsere Vertriebspartner ein höheres Maß an Fachkompetenz für das Beratungs- und Verkaufsgespräch mitbringen sollten, als das in Bezug auf Computer und Anwendungen in dieser Vertriebsschiene in der Regel anzutreffen ist.

Die Zukunft der Sinclair-Computer hierzulande ist nach der Absage von Schneider weiterhin offen. Jürgen Schumpich, bis April 1986 Generalimporteur für Sinclair-Products in Deutschland, steht in Verhandlungen mit Amstrad. Aufgrund seiner 13jährigen Erfahrungen mit Sinclair ist Jürgen Schumpich natürlich stark daran interessiert, mit Amstrad ins Geschäft zu kommen. Wir werden in der nächsten Ausgabe über das Ergebnis der Verhandlungen berichten. *-wt*



Formschön mit integriertem Schwenk-Neigefuß

Für IBM PC und Kompatible wie geschaffen sei der neue Farbmonitor, den Cetec Trading auf den Markt bringt. Die 12"-Bildröhre ist entspiegelt und in einem formschönen Kunststoffgehäuse untergebracht. Der Signaleingang ist RGB-Intensity/TTL positiv. Dieser „Polo" (FTC-12ATH) kann von Farbe auf Grün oder Bernstein umgeschaltet werden und kostet unter 1400 Mark.



Handlicher Ratgeber rund ums Zubehör

Nach dem Motto „in der Kürze liegt die Würze" liefern die einzelnen Karten des Misco-Manuals wichtige Hinweise rund um das Zubehör. Über Bedienungsanleitungen und Einsatzbereiche hinaus gibt diese Lose-Blatt-Sammlung Tips für die verschiedensten Geräte und Produkte. Kostenlos erhältlich bei Misco in 6082 Mörfelden-Walldorf.

BMZ 6283

„Eigenes Geld ohne eigene Bankverbindung? Willste vielleicht auf Papis Konto überweisen lassen?“

An den Kontogebühren jedenfalls soll es nicht liegen, wenn sich Berufsanfänger kein eigenes Konto leisten: Für Azubis ist das KKB-Berufsstartkonto drei Jahre lang gebührenfrei. Das macht sich bezahlt. Und jeder, der 18 ist, kriegt normalerweise auch sofort seine Euroschecks und die Scheckkarte. Das macht unabhängig. Rund um die Uhr.

Sehen wir uns?



KKB Bank *Die Bank für den privaten Kunden.*

Hauptverwaltung: Kasernenstraße 10, 4000 Düsseldorf 1. Die Adresse einer der rund 280 Zweigstellen finden Sie in Ihrem Telefonbuch. Btx *203040 #.



HIT PARADE

Im Auftrag von HC und CHIP ermittelte das Institut Roland Berger & Partner die meistverkauften Home- und Personal-Computer im Juni 1986 (in Klammern wie immer die Plazierung des Vormonats):

Home-Computer

1. Commodore 64 (2)
2. Atari 260 ST (4)
3. Commodore 128 (3)
4. Atari 800 XL (5)
5. Schneider CPC 6128 (6)
6. Schneider CPC 464 (1)
7. Atari 130 XE (7)
8. Philips VG 8010 (–)
9. Commodore 16 (8)
10. Philips VG 8235 (–)
11. Schneider CPC 664 (9)
12. Atari 600 XL (–)
13. Sharp PC 1450 (–)
14. Commodore 116 (–)
15. Sinclair ZX 81

Personal-Computer

1. Atari 520 ST+ (1)
2. Schneider Joyce (2)
3. IBM PC AT (5)
4. Commodore PC 10 (8)
5. Atari 1040 ST (3)
6. Commodore Amiga (4)
7. Tandon PC (–)
8. IBM PC XT (7)
9. Apple Macintosh (6)
10. Commodore PC 20 (–)
11. Apple IIc (12)
12. Triumph Adler P 10 (13)
13. Apple IIe (10)
14. Tandon PCA (–)
15. Tandon PCX (9)

Deluxe-Software für den Amiga

Bereits ausgeliefert werden Deluxe-Paint und Deluxe-Print, das Programm Deluxe-Video soll folgen. Deluxe-Paint als Grafikprogramm arbeitet in allen drei Modi und erlaubt, jede der 4096 Farben zu verwenden. Deluxe-Print und ein grafikfähiger Drucker verwandeln den Amiga in eine Druckmaschine. Karten, Poster und Briefköpfe können entworfen und ausgedruckt werden. Dieses Programm ist kompatibel zu Deluxe-Paint. Deluxe-Video dient zum einfachen Entwerfen und Zusammenstellen von animierten Grafik-Sequenzen. Videofilme können mit Computergrafik versehen und Computer-Videoclips erstellt werden.

SOFT-TOPS

Fußballspiele haben deutliche Tendenz nach unten, FA Cup Football ist auf Platz 21 gefallen. Ein neues Thema begeistert die Spieledesigner: Kernreaktor-Unfälle. ,,Countdown to melt", für C64, ist zwar noch nicht in der Chart, hat aber beste Chancen (aus: Your Computer, London).

1. Bomb Jack	(Elite)
2. Commando	(Elite)
3. V	(Ocean)
4. Green Beret	(Imagine)
5. Formula One Simulator	(Mastertronic)
6. Way Of The Tiger	(Gremlin Graphics)
7. Last V8	(Mastertronic)
8. Comp. Hits 10 Vol 2	(Beau Jolly)
9. They Sold (2)	(Hit Squad)
10. One Man And His Droid	(Mastertronic)
11. Superbowl	(Ocean)
12. Spellbound	(Mastertronic)
13. Finders Keepers	(Mastertronic)
14. Turbo Esprit	(Durell)
15. Action Biker	(Mastertronic)
16. Spindizzy	(Electric Dreams)
17. Starstrike 2	(Realtime)
18. Batman	(Ocean)
19. Yie Ar Kung Fu	(Imagine)
20. Off The Hook	(Off The Hook)

BUCHECKE

Kane, Gerry:
68000 Mikroprozessorhandbuch.
McGraw-Hill-Verlag, Hamburg, 1985; 135 Seiten, 35 Mark.
Kurze Übersicht über die Familie der 68000-Mikroprozessoren. Signale (Hardware) und Befehle (Software) als Grundlage für den Entwurf von 68000-Systemen und deren Implementation.

Lüers/Stein:
Atari ST für Einsteiger.
Data Becker, Düsseldorf, 1986; 262 Seiten, 29 Mark.
Wie man sofort mit dem Atari ST zurechtkommt. Mit fertigen Programmen zum Eintippen, der Programmiersprache Logo und einem Häppchen Computergeschichte.

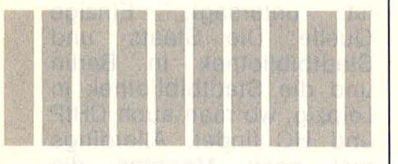
Lüke, I. und P.:
Der Atari 520ST.
Markt & Technik-Verlag, Haar, 1986 (2. erw. Aufl.); 198 Seiten, 49 Mark.
Für Besitzer und Interessierte. Die überarbeitete Auflage trägt den neuen Entwicklungen bei Atari Rechnung, beispielsweise



AT-Power in der Aktentasche

Zwei kompakte Portables von Toshiba.

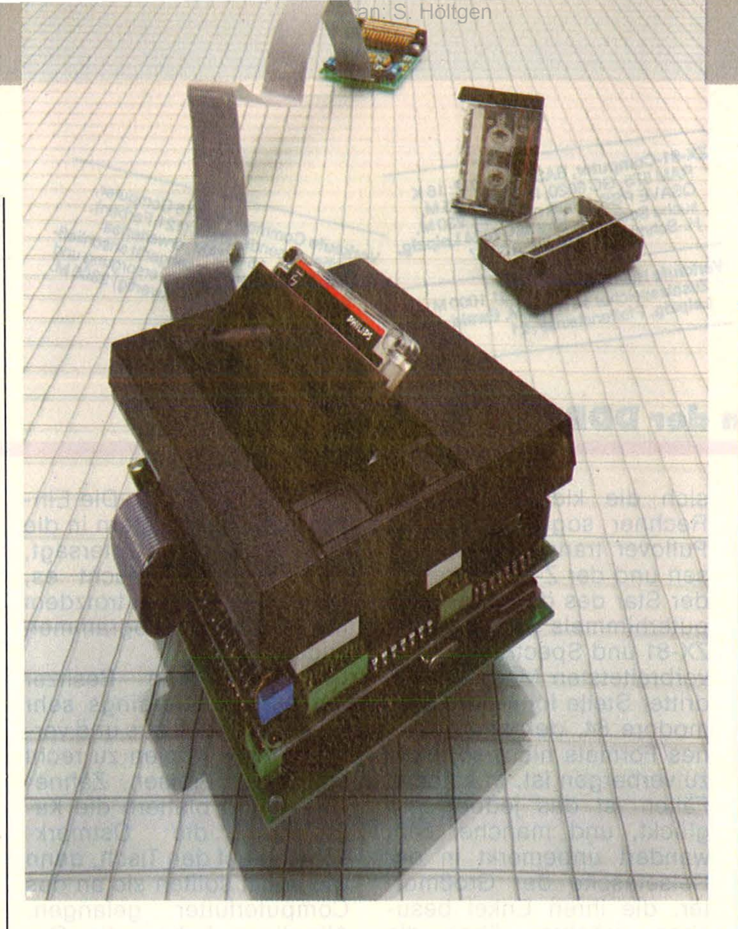
Nicht viel größer als eine Reiseschreibmaschine sind die beiden Portables T2100 und T3100 mit Plasma-Bildschirm. Dazu eingebaute 3,5-Zoll-Diskettenlaufwerke mit 720 KB oder eine Festplatte 256 KB RAM, intern auf 640 KB ausbaubar. Schnittstellen und zusätzliche Erweiterungseinheit vergrößern den Einsatzbereich.



dem deutschen Betriebssystem und geänderten Ausstattungsmerkmalen.

Steiner, J. u. G.:
GEM für den Atari 520ST.
Markt & Technik-Verlag, Haar, 1986 (2. erw. Aufl.); 344 Seiten, 52 Mark.
Alles, was GEM für den Anwender interessant macht: Pull-Down-Menüs, Fenster- und Symboltechnik, Mausbedienung. Für Fortgeschrittene der interne Aufbau von GEM, alle unter diesem Betriebssystem verfügbaren Routinen und wie man sie über C aufruft.

Vieillefond, C.:
Programmierung des 68000.
Sybex-Verlag, Düsseldorf, 1985; 456 Seiten, 150 Abbildungen, 64 Mark.
Für Besitzer von Macintoshs, Sinclair QLs, Atari STs und Commodore Amigas. Der Autor geht sowohl auf die Hard- als auch die Software der 68000-Prozessoren ein. Pinbelegung, Signale, Timing und Peripheriebausteine werden beschrieben, der Befehlssatz wird dargestellt und an Beispielprogrammen erläutert.



Digitaler Kassettenrecorder zur Datenspeicherung

Die analoge Datenaufzeichnung, die bei üblichen Kassettenrecordern am Home-Computer genutzt wird, ist nicht besonders zuverlässig. Für Profi-Anwendungen kommt daher nur die digitale Aufzeichnung in Frage, die allerdings spezielle Laufwerke und Digitalkassetten erfordert.
Für OEM-Anwendungen konzipiert ist der Mini-Digital-Kassettenrecorder MDCR II von Philips, der preislich zwischen Walkman und Floppy angesiedelt ist. Controller und serielles V24-Interface ergänzen das neue Laufwerk. Durch identische Abmessungen können Controller und Laufwerk als „Sandwich" aufeinander gepackt werden.
Der busorientierte Controller kann von Computersystemen jeder Größe angesteuert werden. Handshaking erfolgt hardware- oder softwaremäßig (RS232C-Protokoll), Übertragungsraten zwischen 150 und 19 200 Baud sind wählbar. Die Anbindung sei denkbar einfach, ebenso die Programmierung. Befehle und Statusinformationen werden durch ESCAPE-Sequenzen übermittelt. Die Speicherkapazität beträgt 100 KB pro Spur, in 95 Sekunden ist das Band beschrieben oder eingelesen.

Programm-Generator für Schneider-BASIC

Für faule Anwender drängt sich immer mehr der Gedanke an eine Automation des Programmierens auf. Warum sollen Programme keine Programme schreiben können? Die Tüftler-Union in Österreich (A-6961 Wolfurt) hat einen Programm-Generator für das Schneider BASIC entwickelt, der im Dialog mit dem Benutzer in Erfahrung bringt, was das Programm tun und wie es gestaltet sein soll. Anschließend schreibt er mit Hilfe des eingebauten Syntax- und Algorithmenwissens das angestrebte Zielprogramm in ein File, das geladen und mit anderen Programmen durch „Merge" vereint werden kann. Die erzeugten BASIC-Programme sind frei zugänglich, list- und modifizierbar und ohne Urheberrechtsanspruch (der Urheber ist ja nicht der Mensch, sondern der Generator).

NEWS BOX

Text-Paket für Einsteiger

Ein Textbearbeitungspaket zum Einsteigerpreis bietet Triumph-Adler an: für weniger als 6000 Mark erhält man den IBM-kompatiblen PC Alphatronic P10, das Textprogramm Tex-Ass sowie den Typenraddrucker TRD 7020. Speziell kleineren und mittleren Betrieben wird damit eine Konfiguration angeboten, mit der Briefe, Aufträge und Abrechnungen erstellt sowie Adreßdateien verwaltet werden können.

Farbausdrucke vom Atari ST

Für den Thermotransfer-Drucker TPX-80 hat C.Itoh Electronics ein ROM entwickelt, das Atari ST-Anwendern den Ausdruck von Farbhardcopys im Format DIN A4 ermöglicht.

Ein Drucker für zwei Computer

Der Schnittstellenverdoppler 2C/C/00 von Wiesemann erlaubt, einen Drucker mit Centronics-Eingang gleichzeitig an zwei Computer anzuschließen. Die Umschaltung erfolge automatisch, es sei keinerlei Bedienung oder Software dazu nötig: Derjenige Rechner, der als erster ein Zeichen an den Drucker sendet, sperrt den anderen Computer. Erst wenn einige Sekunden nichts mehr gedruckt wird, erfolgt wieder die Freigabe.

Personal Computer ZX 81 + 16 K RAM-Modul, nur wenige Stunden ben., f. 3100 M zu verk. Bücher: Das ZX 81 Rom, Das ZX 81 Buch je 150 M. **Meinel, 4101 Brachwitz,** Dorfstr. 6

ZX-81-Computer, BASIC, 1050 M; 16 K RAM 575, GC 6020, Interface 675 M, QSAVE PSS, Grafik n. HC. Lit. 230 M, suche Spectrum. **Kirsten, 7024 Leipzig,** H.-Schmidt-Str. 15, Tel. 6 39 97

Verkaufe Heimcomputer ZX 81 1000 M, Zusatzspeicher 16 K 600 M, **Ewald, 7065 Leipzig,** Pfaffensteinstr. 21

Verkaufe Commodore C 16 Computer, hochauflösende Grafik (121 Farben), musikfähig, 16 K RAM, erweitertes BASIC, an jedes Fernsehgerät anschließbar, einschließlich Stromversorgung und Handbuch. Taxwert (neuwertig) 2300 M. **Tel. Berlin 2 82 84 68**

Sinclair ZX Spectrum, 48 K RAM, sowie umfangr. Software, 3500 M. **Tel. Plauen/Vogtl. 3 22 52**

Freaks und Chips in der DDR

Ein harter, mühevoller Weg zum eigenen Computer, Wucherpreise für die Software. Gäbe es nicht die treuherzigen Großmütter aus dem Westen, sähe es für Computerbegeisterte schlecht aus.

Es gibt sie, die Computerszene in der DDR, doch sie hat mit beträchtlichen Schwierigkeiten zu kämpfen. Weder vom eigenen Staat noch von anderen Ländern kann sie Unterstützung erwarten. Die dortige Industrie arbeitet mit dem Z80-Prozessor unter dem Namen U-880. Es gab sogar zwei Home-Computer mit dieser CPU, doch die Auslieferung wurde inzwischen eingestellt.

Was bleibt dem computerinteressierten DDR-Bürger, der nicht die Möglichkeit hat, sich einen Rechner für zu Hause anzuschaffen? Beziehungen in das westliche „Ausland", um sich von dort einen Mikro kommen zu lassen. Doch selbst das ist mit Schwierigkeiten verbunden, denn der Export von Mikrocomputern oder Zubehör in die DDR ist verboten, da sie ja für militärische Aufgaben eingesetzt werden könnten. Man stelle sich einmal einen ZX-81 in einer strategischen Leitzentrale vor! So bleibt nichts anderes übrig, als das Gerät zu schmuggeln.

In den abenteuerlichsten Verstecken werden Computer in das Land geschafft. Da sich die kleinen Sinclair-Rechner sogar unter dem Pullover transportieren lassen und der Z-80-Prozessor der Star des dortigen Computerhimmels ist, gehören ZX-81 und Spectrum zu den verbreitetsten Modellen. An dritter Stelle folgt der Commodore 64, der wegen seines Formats nicht so leicht zu verbergen ist. In etlichen Fällen ist das jedoch geglückt, und mancher 64er wandert unbemerkt in der Reisetasche der Großmutter, die ihren Enkel besuchen möchte, über die deutsch-deutsche Grenze.

Doch die Computerbegeisterten, die nicht mit solch warmherzigen Großmüttern gesegnet sind, möchten auch gerne einen Mikro in der guten Stube haben. Ihnen bleibt nur der Weg des Eigenbau-Modells. Es kann sich über viele Monate, manchmal sogar Jahre, hinziehen, bis der natürlich mit einer Z80-CPU arbeitende Rechner läuft. Denn der Einzelteile-Markt sieht nicht anders aus als der sonstige: Entweder es ist etwas da, oder eben nicht.

Die Hardware-Beschaffung ist nicht das einzige Problem. Was dann immer noch fehlt, ist die Software, die eine Nutzung erst ermöglicht. Da tut sich im Lande eigentlich gar nichts. Außer einem Verbot, Computerprogramme für den privaten Sektor zu handeln. Die Einfuhr von Datenträgern in die DDR ist ebenfalls untersagt, doch wen überrascht es, wenn die Freaks trotzdem mit westlichen Programmen zugange sind.

Die glücklichen Besitzer nutzen das allerdings sehr zu ihrem Vorteil aus und verkaufen ihre Kopien zu recht stattlichen Preisen. Zähneknirschend blättern die Kameraden die Ostmarkscheine auf den Tisch, denn wie sonst sollten sie an das Computerfutter gelangen. Allerdings haben die Geschäftstüchtigen vorher für ihre Originale kräftig berappen müssen. Nicht selten sind Diplomaten ihre Quellen, die aus den „richtigen" Ländern das Handelsgut mitbringen und es zum fünffachen Einkaufspreis — plus einer Transportgebühr — veräußern. Auf dem Schwarzmarkt sind vor allem Spiele wie Pac Man vertreten.

Da hilft nur eins: Man muß sich eigene Programme schreiben. Doch gilt auch hier der allgemeine Wahlspruch der DDR: Wo ein Ziel ist, gibt es auch genügend Hindernisse, um den Weg dorthin zu verbauen. Größte Barriere ist die nicht vorhandene Einsteiger-Literatur. Kein einziges BASIC-Lehrbuch ist öffentlich zu kaufen. Es gibt zwar wenige einführende Schriften, doch sind diese viel zu theoretisch.

Wir, die wir gerade HC lesen, können an jedem Kiosk auf einen Berg von Computer-Zeitschriften zurückgreifen. In der DDR gibt es überhaupt keine. Lediglich zwei Elektronik-Fachblätter haben eine Rubrik „Mikrocomputer". Darunter findet man aber keine Tips und Tricks, sondern liest ausführlich, wie Betriebe ihren Computer einsetzen. Bleibt wieder nur der Lesestoff aus dem Westen. Den gibt es natürlich nicht am Kiosk, auch ein regelmäßiger Bezug aus der Bundesrepublik ist untersagt. Einzige Quelle: Die Staats- und Stadtbibliothek in Berlin und die Stadtbibliothek in Leipzig, wo man auch CHIP und HC findet. Allerdings erst nach Monaten die „neueste" Ausgabe.

Woran die Computerszene in der DDR bislang am meisten krankt, ist die mangelnde Kooperation der Freaks. Wie sollten sie sich auch finden, wenn es weder Kontaktanzeigen in Zeitschriften noch Treffpunkte in Fachgeschäften gibt. Viel Ausdauer und Eigeninitiative gehört zum Computern in der DDR. Deshalb freuen sich die User am meisten, wenn sie von Computerclubs aus dem Westen Tips und Hilfestellungen bekommen. Dabei muß man ja nicht gleich zum Schmuggler werden. *Thomas Tai*



Computer-Display an V.24-Schnittstelle

Für manche Anwendungen genügt es, wenige alphanumerische Daten den Rechner anzeigen zu lassen. Ein Bildschirm ist dann zu groß und benötigt Netzanschluß. Abhilfe schafft die neue LCD-Anzeige, Typ 81000, von Wiesemann in Wuppertal. Die etwa 400 Mark teuren Anzeigemodule werden an eine V24-Schnittstelle des Rechners angeschlossen und sind über Steuercodes adressierbar. Bis zu zehn Stück können gleichzeitig eingesteckt und einzeln angesprochen werden.

Das Display zeigt zwei Zeilen zu je 16 Zeichen, alphanumerisch in Punktmatrix an. Übertragungsrate: 300 bis 9600 Baud, 7 oder 8 Datenbit. Stromversorgung über die Schnittstelle.

Spielend lernen ist ein Wunschtraum geplagter Schüler. Doch ein Computer ist keine Wundermaschine



Der elektronische Pauker

Eine Zeitlang sah es so aus, als ob der ,,Braunschweiger Trichter" (nach der Produktionsstätte eines weit verbreiteten Home-Computers) alle Lernprobleme beseitigen könnte: Als nämlich neben den Geschicklichkeits- und Ballerspielen eine reiche Auswahl an Lernspielen auf den Markt kam, die den schwachen Schülern den Nachhilfelehrer ersetzen sollte.

Endlich gab es ein gewichtiges Argument für die stundenlange Beschäftigung mit diesen Kisten. Und die Erziehungsverpflichteten sahen es mit Wohlwollen, denn schließlich sollen er oder sie etwas fürs Leben damit lernen. Fast jeder, so steht es ja in den Zeitschriften, wird in Zukunft mit Computern zu tun haben. Doch langsam werden auch Zweifel laut, ob der schwanzwedelnde Löwe oder die angriffslustigen Vokabeln wirklich dem Bildungsdrang so förderlich sind oder nur ein weiteres Spiel darstellen und vom eigentlichen Lernstoff ablenken.

Daß das Spielen ein wichtiger Bestandteil unseres Lebens ist, kann niemand bestreiten. Auch nicht, daß es zum wirkungsvollen Lernen gehört. Allerdings dienen spielerische Elemente in unseren Bildungsstätten nur vordergründig der Auflockerung. Dahinter steckt die Leistung, die zu erbringen ist — und die letztendlich zu einer Bewertung, einer Note führen soll. Beispiele dafür: Zahlenspiele, Städte raten, Vokabeln im Wettstreit üben. An diesen Vorbildern orientieren sich die klassischen Computer-Lernspiele. ,,Belohnt" werden richtige Antworten durch Melodien, die der Rechner von sich gibt, oder durch bewegte Grafiken: Männchen wandern eine bestimmte Strecke vorwärts, Punkte werden aufgesammelt oder Fehlerteufel vernichtet. Ist dabei das Spiel zu sehr betont, wird der Lernstoff selbst zur Nebensache. Fraglich ist, ob das nützt oder nur eine Anbiederung an spielgewohnte Freaks darstellt.

Lernspiele bergen auch die Gefahr in sich, daß der Lernende auf einen reinen Maschinen-Dialog trainiert wird, der aus dem Drücken von Tasten oder Joystick-Bewegungen besteht. Damit kommt eine wichtige Aufgabe des Lernens zu kurz, die soziale Erziehung, die Ich-Erfahrungen in der Gemeinschaft, die Team-Arbeit, Toleranz und Gemeinschaftssinn.

Ausschließliches Lernen mit Computerprogrammen darf und wird es auch nicht geben. Wer längere Zeit vor dem Bildschirm sitzt, wird dies als sehr belastend empfinden. Der leichte Zugriff auf gespeicherte Informationen darf nicht dazu führen, daß nur noch diese vermittelt werden — und nicht



mehr die Beobachtung und Auswertung der Wirklichkeit, von Experimenten (leider ist bei vielen Wissenschaftlern dieser Trend zu sehen). Schließlich werden wir von Tag zu Tag ärmer an ursprünglicher Erfahrung. Wie unsere Kommunikation sich unter dem Einfluß der Arbeit mit dem Computer ändern wird, können wir im Augenblick nicht absehen. Die einfach strukturierte Dialogsprache der Programme könnte unsere sprachliche Entwicklung prägen.

Doch schließlich entstehen mit der neuen Generation an Home-Computern mit mehr Speicher und besseren Grafikfähigkeiten auch neue Möglichkeiten für Programmkonzeptionen, die der „künstlichen Inteligenz" ein Stück näher kommen. Zumindest theoretisch werden Programme möglich, bei denen der Computer nicht mehr stumpfsinnig Wissen abfragt und mit einfältigen Antworten wie „Richtig – gratuliere!" oder „Falsch – versuche es noch einmal" kommentiert, sondern gezielt auf das Wissen des Lernenden reagiert, um beispielsweise Denkanstöße zu geben, Vergleiche heranzuziehen, Wissen aus anderen Gebieten einfließen zu lassen. Eine aktive Auseinandersetzung mit dem Stoff wäre das Ziel. Möglich wäre es, wenn sich der Computer wie ein idealer Lehrer verhielte: Wenn er den Lernenden ermutigen, ihm nicht Verstandenes auf andere Weise verdeutlichen könnte, wenn er – mit Hilfe von Bildplatten, CD-ROM und anderen interaktiven Systemen – den Stoff mit Grafiken und Bildern untermalen, also von selbst Hilfsmittel bereitstellen könnte, die zur Vermittlung von Lerninhalten nötig sind.

Mit solchen Programmen stünden nicht nur intelligente „Lehrer" zur Verfügung, sondern auch die Möglichkeit, das Wissen über moderne Lehrstrategien aus vielen Quellen zusammenzufassen, ähnlich der „künstlichen Intelligenz" auf anderen Gebieten, die vorerst auch nichts anderes ist als Wissensansammlung von Experten für denjenigen, der dieses Wissen nicht hat. Doch damit wird der Lehrer aus Fleisch und Blut keineswegs arbeitslos! Lernen ist schließlich nicht nur Aufnehmen von Fakten. Die Gemeinschaft der Lernenden, der Wettbewerb in der Gruppe, der Ansporn, das Erfolgserlebnis läßt sich am Computer nicht nachvollziehen. *-reh*

Textprogramme helfen gegen Schreibschwächen

Erfahrungen aus der Praxis: So kann richtig Schreiben und Lesen geübt werden

Mit „Legasthenie" bezeichnet man eine Lese- und Rechtschreibeschwäche, die viele Ursachen haben kann. Ein Schüler schreibt beispielsweise „Hoite ist es ser hais" – was der üblichen Sprechweise entspricht. Wie kann ihm geholfen werden?

Fehl am Platz sind starre Paukprogramme und einseitige Diktatübungen. Im Vordergrund sollten Aufgabenstellungen stehen, die ein selbständiges Arbeiten mit Texten zum Ziel haben. Seit fast 15 Jahren trainieren wir auf vielfältige Weise mit dieser Zielgruppe und machen seit Monaten die besten Erfahrungen beim Einsatz von Textverarbeitungsprogrammen.

Allerdings ist bei der Anschaffung auf folgende Punkte unbedingt zu achten:

∗ Groß-/Kleinschreibung möglich?

∗ Umlaute (ä,ö,ü) vorhanden?

∗ „ß"-Schreiben da?

Leider sind diese Mindestanforderungen sogar bei speziellen Deutschprogrammen nicht immer verwirklicht (zum Beispiel nicht

Programme für jeden Zweck

Zu fast jedem Lernstoff gibt es inzwischen Programme. Hier ein Überblick über Themen und ihre Anbieter – ohne Anspruch auf Vollständigkeit.

Fremdsprachen
Data Becker (für C64);
Ernst Klett Verlag (für Apple II);
Intus Lern-System (für Apple);
Langenscheidt (für C64);
Markt & Technik (für C64);
Max Hueber Verlag (für C64, ZX Spectrum);
SM Softtraining (für C64);
Westermann Schulbuchverlag (für C64);
Wilhelm Hagemann (für C64, VC 20).

Deutsch (Grammatik und Rechtschreibung)
Atari GmbH (für Atari 400/600/800 XL);
Intus Lern-System (für Apple II);
Westermann Schulbuchverlag (für C64).

Mathematik
Bayerischer Schulbuch-Verlag (für C64);
Data Becker (für C64);
Ernst Klett Verlag (für C64);
Intus Lern-System (für Apple II);
Sybex Verlag (für C64);
Westermann Schulbuchverlag (für C64);
Wilhelm Hagemann (für C64, VC 20).

Biologie, Chemie, Physik
Intus Lern-System (für Apple II).

Schreibmaschinenkurs
Ariola Eurodisk (für C64);
Atari GmbH (für Atari 600/800 XL);
Intus Lern-System (für Apple II);
Triumph Adler (für TA alphatronic PC)

beim „Rechtschreiblöwen" von Westermann). Sie kommen dann für ein Training zur Überwindung der Lese- und Schreibschwäche nicht in Frage.

Wünschenswert wären außerdem leichtes Erlernen und einfache Handhabung, Hervorhebung einzelner Wörter oder Merksätze durch Fettdruck oder Unterstreichen sowie vielseitige Anwendungsmöglichkeiten.

Welche Vorteile bieten die Textverarbeitungsprogramme? Sie gestatten das Lernen an frei gestalteten Texten (dies ist wichtig zur Identifikation, denn fremde Texte interessieren den Lernenden weniger). Sie ermöglichen das Erstellen gezielter Lückentexte, in die Wortteile oder ganze Wörter einzusetzen sind. Sie erlauben den Entwurf von Übungsblättern und später auch deren Ausgabe auf den Drucker. Wichtig, wenn man bestimmte Wortteile oder Regeln hervorheben möchte. Sie verlangen und fördern eine sehr genaue Arbeitsweise. Sie bieten Korrekturmöglichkeiten schon am Bildschirm, ehe ein Text zu Papier gebracht wird – das Gesamtschriftbild bleibt sauber!

Lesen und Schreiben per Computer unterstützt durch die Druckschrift eine bessere Durchgliederung des Wortbildes und vermeidet durch die Eingabe von Einzelbuchstaben das für Legastheniker oft unangemessene Schreibtempo der Schreibschrift. Das gilt übrigens auch für die Arbeit mit einer ganz normalen Schreibmaschine, bei der man aber keine Fehler so einfach beheben kann.

Und schließlich erhält jeder Anwender eines Textverarbeitungsprogramms eine erste Einführung in die EDV, was auch zu schätzen ist. Außerdem werden Lernende am besten durch Erfolgserlebnisse zur Weiterarbeit motiviert.

Unsere Vorschläge: Wegen der guten Erlernbarkeit (bereits zwölfjährige Schüler haben damit gearbeitet) empfehlen wir das weit verbreitete, allerdings auch etwas teure Textverarbeitungsprogramm „Vizawrite". Etwas Hilfestellung zum Verständnis des umfangreichen Handbuchs ist jedoch ratsam. Ebenfalls in Frage kommt „Startexter" (Sybex). Das Handbuch ist verständlich geschrieben, mit der Version 4.0 läßt sich gut arbeiten.



Foto: Rüdiger Werner

Lernen am Computer: *Fünf Commodore im Verbund*

Lernsoftware – aus der Praxis entwickelt

Nicht am grünen Tisch, sondern in der Schule wurde ein Programm entwickelt, das Lehrern die Arbeit erleichtern soll

Über einen Multi-User-Zusatz und eine Floppy 1541 steuert der Schweizer Realschullehrer Armin Rosenast sechs C64-Computer in seinem Klassenzimmer. Was dabei herauskommen kann, zeigt eine neue Software mit dem Titel „Lernkartei". Es handelt sich hierbei um eine Lernsoftware, die in dieser Form neu ist. Nicht nur ein Trainingsprogramm für Rechtschreibung ist damit gegeben, sondern auch eine elektronische Grammatik. Darüber hinaus enthält die „Lernkartei" auch noch Erdkunde, Biologie und mehr.

Sie bietet weiterhin ein Eingabeprogramm zur Erstellung eigener Lerndateien. Zusätzliche Leistungen sind Diskettenoperationen wie Formatieren und Druckeransteuerung.

Als Besonderheit aufgefallen: Aus den Dateien können einzelne Fragen zusammengestellt und auf diese Weise ganze Prüfungsbogen entwickelt werden. Wem das noch nicht genügt, der kann sich ein Arbeitsblatt ausdrucken lassen und mit Hilfe des unsichtbaren Codewortes „schubi" die dazugehörenden Lösungen abheften. Klassenarbeiten lassen sich auf diesem Weg besonders zeitsparend erstellen.

Derzeit gibt es dieses Programm für den C64 auf Diskette für etwa 90 Mark beim Huesmann + Benz Verlag, Singen. Eine Schneider-Version und Datendisketten zu weiteren Sachgebieten seien in Arbeit.

Rüdiger Werner

Bewertung

Durch gute Menüsteuerung ist die Arbeitsweise nach kurzer Einarbeitungszeit weitgehend selbsterklärend. Rücksprung zum Hauptmenü ist fast immer möglich. Die Umlaute ä, ö, ü sind vorhanden, das „ß" fehlt noch, soll aber in der neuen Version berücksichtigt werden.

Das Anleitungsheft ist klar verständlich geschrieben, die zahlreichen Beispiele machen die Arbeit recht einfach.

Der in der Vorabversion verwendete Wortschatz und die Grammatikübungen sind noch nicht in allen Fällen an den bei uns üblichen Standardwortschatz angeglichen, um als typisches Schulprogramm zugelassen zu werden. Die Neufassung will unsere Verbesserungsvorschläge berücksichtigen.

Im Hinblick auf die vielfältigen Einsatzmöglichkeiten ist das Preis-Leistungs-Verhältnis akzeptabel.

T.S. Datensysteme-Vertriebsgesellschaft mbH

Das sind Preise!!!

DISCOVERY 180
ein Laufwerk
180K-Speicherplatz
DM 399,–

DISCOVERY 360
zwei Laufwerke
360K-Speicherplatz
DM 599,–

DISCOVERY 720
ein Laufwerk
720K-Speicherplatz
DM 799,–

DISCOVERY 1400
zwei Laufwerke
1,4M-Speicherplatz
DM 1199,–

DISCOVERY PLUS 18
Einbausatz mit
1 Zusatzlaufw. 180K
DM 199,–

DISCOVERY PLUS 72
Einbausatz mit
1 Zusatzlaufw. 720K
DM 399,–

BIS ZU 1,4 MBYTE FÜR JEDEN SPECTRUM

Die neuen Discovery-Systeme mit – Centronics-Drucker-Interface – Joystick-Interface (Kempston-Typ) – Video-Monitor-Ausgang – stabilisiertem Netzteil (versorgt auch Ihren Spectrum) – Befehlssatz voll Microdrive kompatibel – belegt keinen RAM im Spectrum – deutsches Handbuch – 1/2 Jahr volle Garantie

Saboteur

Der Topseller aus GB für Spectrum und C-64 29,90

MULTIFACE ONE

Endlich ist das Problem der Sicherungskopien endgültig gelöst. Mit Multiface können Sie **jedes** Programm, das Sie in Ihren Spectrum geladen haben, auf Band, Microdrive Cartridge oder Floppydisk abspeichern. Daneben ist eingebaut: Joystick-Interface, Composite Video-Interface, SKB ROM, SKB RAM, den Sie auch sonst nutzen können. Multiface speichert den gesamten Rechnerinhalt (inkl. Register) auf jedes am Spectrum anschließbare Medium. Natürlich haben Sie 6 Monate Garantie.

DM 159,90

Kempston PRO

Das wohl umfassendste Joystick-Interface für den Spectrum. Insgesamt drei Anschlüsse (davon 2 wie Interface 2, zusätzlich Kempston- und Cursor-Key-Interface) natürlich auch mit ROM-Slot für Sinclair-Module **59,90**

Samantha Fox Strip-Poker

Bei diesem neuen und wirklich aufregenden 7 Card Stud Poker müssen Sie schon ganz schön cool bleiben. Ob Sie es schaffen, Ihre Spielkarten stets im Auge zu behalten? Digitalisierte Videos – und bluffen kann Samantha auch.

Für Spectrum 48K/128K	29,90
Für Schneider 464/664/6128	29,90
für Schneider (Disk)	39,90
für C-64 (Disk.)	39,90
für C-64 (Cass.)	29,90

Drucker-Interface

Kempston Typ E-Interface mit Software im ROM, d.h. es wird kein Speicherplatz belegt. Routinen für alle gängigen Druckertypen bereits implementiert. COPY- und Vierfachcopy möglich.

Phantastischer Preis **179,90**

Utilities für den Spectrum

HISOFT-Pascal mit deutscher Anleitung	99,90
HISOFT-Basic-Compiler COLT mit deutscher Anleitung	59,90
HISOFT-C-Compiler, die Programmiersprache der Zukunft	99,90
HISOFT-Devpac-Assembler und Disassembler im Paket	59,90
Bels Basic 3.0 (deutsche Version auch für Discovery)	59,90

C 16-Software

ACE	35,90
The Berks Trilogy	25,90
Killapede	9,90
Basseball	25,90
DT Star Events	25,90
Legionnaire	25,90
Petch	14,90
Lawn Tennis	25,90
Airwolf	29,90
Gunslinger	19,90
Bongo	19,90
Pogo Pete	19,90
Flight Path 737	19,90
Kung Fu Kid	19,90
Atlantis	14,90
Hustler	19,90
Reach for the Sky	19,90
Slipery Sid	14,90
Classic II (4 Spiele)	29,90
Thai Boxing	19,90
Star Commander	14,90
Commando (Original)	25,90
Joystick-Adapter	8,90
Grand Master Chess	34,90
Gullwing Falcon	29,90
Bomb Jack	24,90
Frank Bruno Boxing	24,90

DISKETTEN $3^1/2''$

Markenqualität **mit** Garantie auch doppelseitig verwendbar **6,99**

C-64

The Way of the Tiger (Disk.)	45,90
The Way of the Tiger (Cass.)	35,90
Nexus	35,90
Ghosts'n Goblins	35,90
Space Invasion	35,90
Law of the West	35,90
Bounces	35,90
Biggles	35,90
Couldron II	35,90
Saboteur	29,90
The Comet Game	35,90
Green Beret	35,90
Rupert Ice Castle	29,90
Empire	35,90
Super Bowl	29,90
Sam Fox Strip Poker (Disk.)	45,90
Ballblaser	35,90
The Planets	35,90
„V"	29,90
Psi-5 Trading Company	35,90
War Play	29,90
Pyjamarama	29,90
Automania	29,90
Choplifter	19,90
Combat Lynx	35,90
Titanic	35,90
Spelunker	19,90
Critical Mass	35,90
Enigma Force	35,90
ACE	39,90
Breakdance (Disk.)	29,90
Laser Basic (Cass.)	39,90
Laser Basic (Disk.)	49,90
Laser Compiler (Cass.)	49,90
Laser Compiler (Disk.)	59,90
Everyone's a Wally	29,90
Herbert's Dummy Run	29,90
Blade Runner	29,90
The Rocky Horror Show	29,90
Space Doubt	29,90

Sinclair QL

QL-Hyperdrive (3-D-Autorennen)	59,90
QL-Match-Point (3-D-Tennis)	59,90
QL-Chess (Schachprog. neue Version)	79,90
Spook	49,90
Zapper	49,90
APL (Keyword)	399,90
BJ-In 3D-Land	49,90
ICE Cholce	59,90
Supercharge (Compiler)	199,90
R Windows	49,90
Key Define	49,90
Assembler	119,90
Block Control	59,90
Mailing List	59,90
Superforth + Reversi Vl.3	119,90
Reversi	49,90
Meteor Storm	49,90
Integrated A/Cs (Sagesoft)	299,90
Macro Assembler (Sinclair)	199,90
Super Astrologer	99,90
Q-Draw	59,90
Archiver	75,90
Compendium (Disk.)	119,90
Toolkit II (ROM)	159,90
BCPL	199,90
ICE Compendium	199,90
HISOFT Mon QL	99,90
HISOFT Superbasic	99,90
Englisch Trainer	59,90
Lands of Havoc	69,90
Scrabble	49,90
QL Super Sprite Generator	49,90
Cosmos	59,90
Knight Flight	59,90
Area Radar Controller	49,90
West	49,90
Gitadel	49,90
The Lost Kingdom of Zkul	49,90
QL-Jabber	49,90
Quazimodo	59,90
QL-Pawn	74,90
Bridge-Player II	59,90
Star Guard & Galactic Invaders	59,90
Night Nurse	54,90
Catridge Doctor	59,90
Kempston Diskinterface (Shugart Bus)	399,00
Laufwerk $3^1/2''$ kompl. mit Netzteil	499,00
Kempston Centronics Interface (ROM)	159,90

UNGLAUBLICHE PREISE

(solange Vorrat) für Spectrum

Computer-Hits 10 Teil 1	19,90
Computer-Hits 10 Teil 2	22,90
Mounty on the Run	14,90
They sold a Million	22,90
Winter Games	19,90
Surf Champ (inkl. Surfbrett)	14,90
Zorro (Originalspiel!)	17,90
William Wobbler (Tophit!)	15,90
Fighting Warrior	15,90
Ping-Pong	19,90
Night-Shade (von Ultimate)	23,90

Brandneue Spectrum-Software

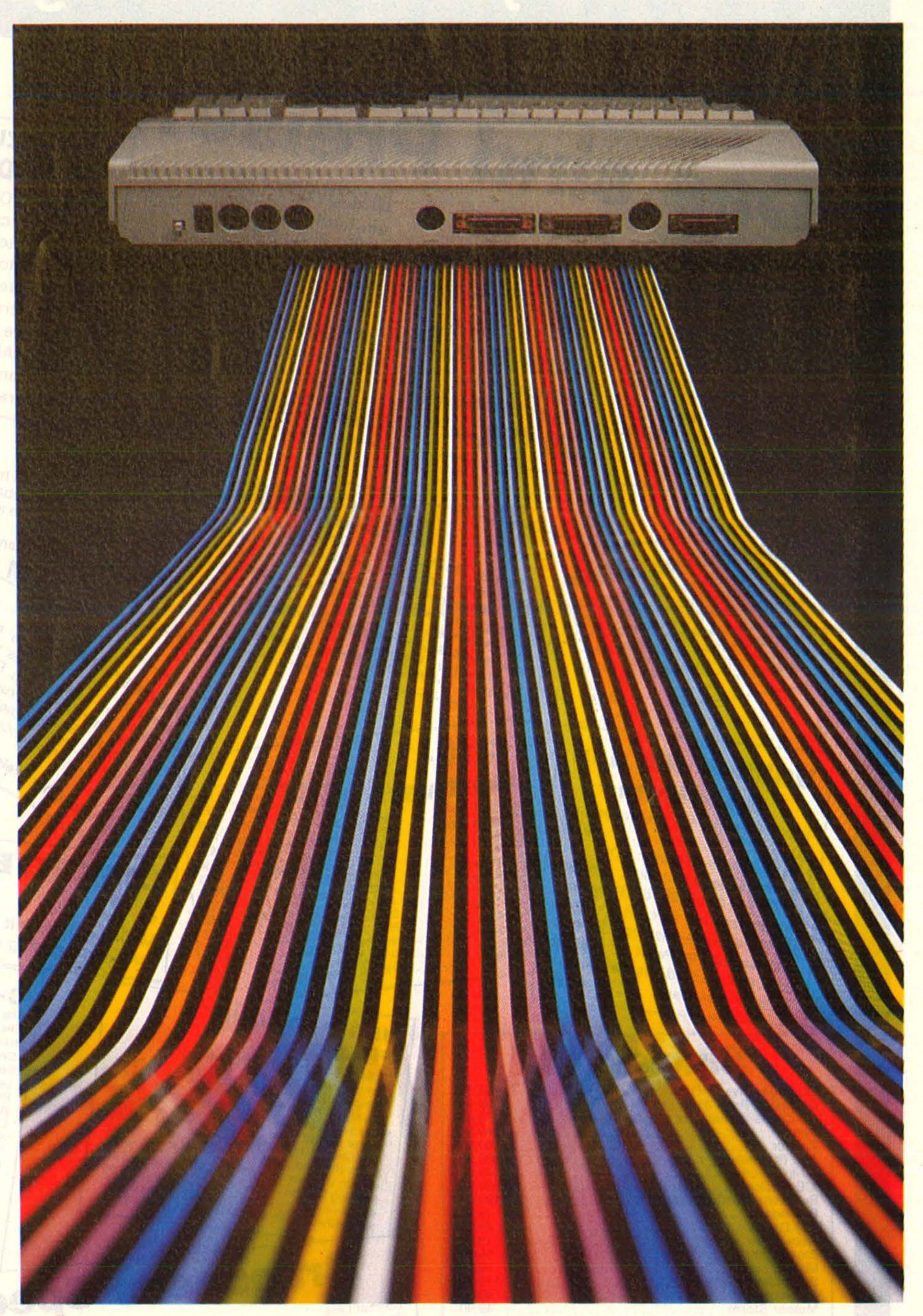
Combat Lynx	29,90
Flyer Fox	19,90
Sal Combat	29,90
The Great Fire of London	19,90
Three Weeks in Paradise	35,90
Yabba Dabba Doo	29,90
Spitfire 40	35,90
Supersleuth	29,90
Critical Mass	29,90
Death Wake	29,90
Pud Pud	25,90
Tomahawk	34,90
Fighting Warrior	24,90
Rambo (Superpreis!)	19,90
Confuzion	24,90
Bulls Eye	24,90
Back to Skool	24,90
Starion	24,90
Atlantis	24,90
Falcon Patrol II	19,90
Raspitin	29,90
Vectron	29,90
Battle of the Planets	29,90
Turbo Esprit	29,90
Super Bowl	29,90
Saboteur	29,90
„V"	29,90
2112 AD	29,90
Gunfight	39,90
Twister	29,90
Pyjamarama	25,90
Schizofrenia	29,90
Green Beret	29,90
Heavy on the Magick	35,90
Art Studio	49,90
Superpipline II	29,90
Batman	29,90
Costa Capers	29,90
Commando (Originalspiel)	29,90
Bounces	35,90
Cauldron	29,90
Movie	29,90
Hocus Focus	29,90
Mantronics	29,90
Way of the Tiger	29,90
Muggsy's Revenge	29,90
Arcade Hall of Fame (4 Spiele US Gold)	35,90
The Young Ones	29,90
Laser Compiler	29,90
Laser Basic	39,90
Yie ar Kung Fu	29,90
Basketball	29,90
Tau Ceti	29,90
Blade Runner	29,90
The Rocky Horror Show	29,90
Juggemaut	29,90
ID	29,90
Rebel Planet	35,90
Arc of Yesod	29,90
Red Hawk	29,90
Rock'n Wrestle	29,90
Toad Runner	29,90
Max Headroom	29,90
Valkyrie	29,90
Spindizzy	35,90
Runestone	29,90
Red Arrows	29,90
Ghosts and Gonlins	29,90

Versand per Nachnahme (zuzüglich 5,90) oder Vorkasse mit Scheck (zuzüglich 2,50)
Informationen können angefordert werden. Händleranfragen erwünscht.

Denisstraße 45, 8500 Nürnberg 80, Tel. 09 11/28 82 86

Bevor man die vielfältigen Möglichkeiten des Computers und der verschiedenen Peripheriegeräte nutzen kann, muß zunächst einmal eine Verbindung zwischen der Konsole und dem angeschlossenen Zusatzgerät hergestellt werden. Hierbei kommt es nicht selten zu Schwierigkeiten, weil die Hersteller von Computern daran interessiert sind, dem Anwender Peripherie aus dem eigenen Haus zu verkaufen. Aus diesem Grund werden viele Anschlußstellen vorgesehen, in die artfremde Geräte nicht ohne weiteres Eingang finden. So ergeben sich für den Computerbesitzer die Fragen, ,,Welches Gerät kann ich anschließen?", ,,Wie wird eine einwandfreie Datenübertragung erreicht?" und ,,Was ist zu tun, wenn die Geräte unterschiedliche Anschlußstellen besitzen?".

Bei Schnittstellen (engl. Interfaces) kann man zunächst einmal zwischen zwei Grundtypen unterscheiden: Schnittstellen, die im Computer beziehungsweise Peripheriegerät integriert sind, also Anschlußbuchsen für Laufwerk, Drucker, Monitor, Joystick, Maus und Modem, sowie Schnittstellen, die zur Datenkommunikation zwischen zwei Geräte geschaltet werden. Hierbei handelt es sich meist um Schnittstellenwandler, die zum Beispiel die ausgehenden Daten eines Computers für einen Drucker mit anderem Anschlußsystem konvertieren, das heißt in einen für den Drucker verständlichen Code umwandeln. Genaugenommen dienen solche Interfaces also zur Datenübertragung zwischen Schnittstellen. Dank dieser Bauteile, die häufig von kleineren privaten Unternehmen angeboten werden, ist es möglich, Computer und Peripherie mit unterschiedlichen Anschlüssen zu verbinden. Unabhängig von der Art der Schnittstelle



PFADFINDER FÜR PERIPHERIEDATEN

Große Schnittstellen-Übersicht für Atari, Commodore, IBM PC und Kompatible, Schneider sowie Sinclair

und dem verwendeten Übertragungssystem bestehen alle Daten, die zwischen zwei Geräten übertragen werden, aus Binärinformationen. So stellt zum Beispiel der Buchstabe „A" nach dem vielgebrauchten ASCII-Code die Binärinformation 1000001 dar. Auch andere Informationen, etwa spezielle Ausdruckskommandos, setzen sich aus Binärdaten zusammen. Bei der Datenübertragung werden diese Codes in elektrische Impulse umgewandelt und über Datenleitungen gesendet. Dabei erhält eine „1" eine andere Stromspannung als die „0". Obwohl es bei der Form der Schnittstellenanschlüsse zum Teil große Unterschiede gibt, haben sich bei der Übertragung solcher Binärdaten zwei Grundprinzipien weltweit durchgesetzt: die serielle und die parallele Datenübertragung.

Serielle Datenübertragung

Die am weitesten verbreitete Form der seriellen Datenübertragung findet sich in der RS232C-Schnittstelle (auch V.24-Schnittstelle). Hinsichtlich der Pinbelegung und Bedeutung der übertragenen Signale wurde von der IOS (International Organization for Standardization) die RS232C-Schnittstelle unter dem Standard 2110 international genormt. Dieser Standard entspricht in Deutschland der DIN 66020.

Bei der RS232C-Schnittstelle werden die Daten seriell, also hintereinander auf einer Datenleitung, übertragen. Zu Beginn des Transfers wird ein Startbit mit einer Spannung zwischen +3 und +15 Volt zum angeschlossenen Gerät gesendet. Danach folgt der Binärcode des Zeichens oder Kommandos, wobei eine „1" einem Stromimpuls zwischen −3 und −15 Volt und eine „0" einem Impuls zwischen +3 und +15 Volt entspricht. Als Abschluß folgen 1, 1½ oder 2 Stoppbits. Sie übernehmen die Funktion einer Pause. Mit der RS232C-Schnittstelle lassen sich Datenübertragungsgeschwindigkeiten von 300 bis 19 200 Impulsen pro Sekunde erreichen. Das entspricht rund 37 bis 2400 Textzeichen oder anderen Einzelinformationen. Zum einwandfreien Betrieb einer RS232C-Schnittstelle müssen eine ganze Reihe von Einstellungen vorgenommen werden.

Parallele Datenübertragung

Die weltweit häufigste Parallelschnittstelle wurde von der amerikanischen Firma Centronics entwickelt und ursprünglich nur in Druckern aus dem eigenen Hause eingebaut. Doch dank der ausgereiften Technik dieses Systems wurde die Centronics-Schnittstelle von vielen anderen Herstellern übernommen.

Im Gegensatz zur RS232C-Schnittstelle erfolgt hier die Übertragung des Binärcodes auf acht Datenleitungen gleichzeitig, das heißt parallel. Aus diesem Grund spricht man bei der Centronics-Schnittstelle auch von der 8-bit-parallel-Schnittstelle. Mit diesem System können wesentlich höhere Übertragungsgeschwindigkeiten erreicht werden. Baud-Raten von 150 KByte bis 1 MByte bei hochwertigen Geräten sind heute möglich. Das entspricht etwa 19 000 bis 131 000 Zeichen oder Einzelinformationen pro Sekunde. Die Stromspannung für binär „0" liegt zwischen 0 bis 0,4 Volt und für binär „1" zwischen 2,4 und 5 Volt.

Neben den acht Datenleitungen kann die Centronics-Schnittstelle noch zahlreiche andere Datenleitungen aufweisen, die jeweils verschiedene Funktionen übernehmen und nicht standardisiert sind. So gibt es zum Beispiel zwei Handshake-Leitungen, die die aktuellen Betriebszustände der verbundenen Geräte signalisieren.

Schneider CPC: Vielfältige Möglichkeiten

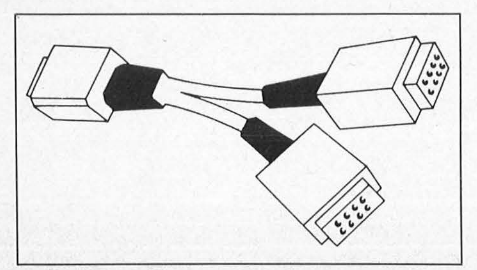
Der CPC 464 sowie die Nachfolgemodelle bieten eine Mischung aus genormten und ungenormten Schnittstellen. An die Stromversorgungsbuchse läßt sich neben dem Versorgungsstecker aus dem Monitor noch ein externes Netzteil namens „MP-1" anschließen. Dadurch wird der Betrieb zum Beispiel mit einem herkömmlichen Farbfernsehgerät möglich, was sich speziell für CPC-Besitzer anbietet, die lediglich einen Grünmonitor betreiben und Grafiken oder ähnliches in Farbe sehen wollen. Das MP-1 besitzt seinerseits die Verbindungen zum Monitoreingang und einen Ausgang zum Fernsehgerät (normaler Antennenstecker). Die Monitorbuchse ist sechspolig für RGB (Rot, Grün, Blau), Luminance (Helligkeit), Synchronisation und Erdung. Die optimale Bildqualität wird zweifellos mit einem Monitor erreicht. Beim Anschluß eines Fernsehers muß eine gewisse Unschärfe in Kauf genommen werden; die 80-Zeichen-Darstellung ist nur schwer lesbar. Leider trifft dies auch auf den Original-Farbmonitor von Schneider zu, der bei bestimmten Farbkombinationen kein befriedigendes Bild liefert.

Genormt ist die neunpolige Joystick-Buchse mit ihren typischen, leicht angeschrägten Ecken, die den Anschluß von praktisch jedem marktüblichen Joystick erlaubt. Der Pferdefuß dabei: Schneider hat den CPC mit lediglich einer Joystick-Buchse versehen, eine zweite befindet sich an dem hauseigenen Joystick „JY-1". Wer zwei Joysticks benötigt (zum Beispiel für Computerspiele) und nicht den relativ teuren JY-1 kaufen will, kann auf den freien Markt ausweichen und den Joystick-Adapter von Dynamics anschaffen und zum Beispiel zwei Joysticks des Typs Competition Pro „Chief" anschließen, die drei Feuerknöpfe besitzen. Als Druckeranschluß bietet der Schneider eine abgemagerte Centronics-Schnittstelle. Abgemagert deshalb, weil zwei Stifte (Pins) weniger verfügbar sind als üblich (18 und 36). Aus diesem Grund ist es zwar möglich, jeden normalen



Drucker mit genormter Centronics-Schnittstellenbuchse anzuschließen, jedoch benötigt man ein spezielles Druckerkabel, das der Ausgangsseite der Buchse des CPC entspricht.

Mit ein klein wenig Geschick ist es allerdings leicht möglich, ein solches Kabel selbst herzustellen, wenn man ein 36adriges Flach-



Spielfreudig: *Die fehlende zweite Joystick-Schnittstelle läßt sich durch einen Adapter beheben*

bandkabel, einen „Double-Row"-Stecker mit 34 Positionen (zum Beispiel von Misco) und einen gewöhnlichen Centronics-Stecker kauft. Diese Teile werden nicht gelötet, sondern durch einen Klemmverschluß Eins-zu-Eins verbunden; das heißt: Pin 1 am Ausgangsstecker muß mit Pin 1 am Eingangsstecker korrespondieren usw. Die Pins 18 und 36 werden ausgelassen.

Ein besonderes Problem bei der Schnittstellenverbindung zu einem Drucker ergibt sich beim Schneider CPC durch den Umstand, daß Pin 14 vom Rechner aus mit Impulsen versorgt wird, die bei den meisten Druckern einen Zeilenvorschub bewirken. Dadurch entstehen beim Ausdruck stets Leerzeilen. Man kann diese Verbindung entweder ablöten oder – wenn man, wie oben gesagt, ein Druk-



Für das Joystick-Interface: *Mit drei Feuerknöpfen gehört der Competition Pro Chief mit zum Feinsten*

kerkabel selbst herstellt – die Verbindung zu Pin 14 auslassen.

Zu den wichtigsten Schnittstellen des CPC zählt zweifellos die 50polige Erweiterungsleiste, die den Computer mit einem externen Diskettenlaufwerk, Modem oder RAM-Erweiterungen verbindet (jedoch nicht alles gleichzeitig). Besitzer des CPC 464 wissen, daß zusammen mit dem 3-Zoll-Laufwerk auch das Disketten-Interface „DDI-1" geliefert wird. Dieses Interface wird auf die Erweiterungsleiste gesteckt und stellt im Kern den von NEC, Rockwell und Intel vertriebenen Floppy-Disk-Controller (FDC) µPD 765 dar. Mit dem µPD 765 ist es möglich, die gesamten Leistungen des Diskettenbetriebs zu vereinfachen und zusätzlich in der Geschwindigkeit zu steigern. Systeme ohne ein vergleichbares Interface (zum Beispiel die Commodore-Diskettenstation 1541) weisen einen höheren Konstruktionsaufwand vor, arbeiten langsamer und sind umständlicher zu programmieren. Der FDC des Schneider belegt 40 oder 50 Pins und stellt alle Signale zur Verfügung, die für die handelsüblichen Diskettenformate 8, 5¼, 3½ und 3 Zoll benötigt werden. Aus diesem Grund ist es theoretisch möglich, eine Schneider-Diskette in einem Format einzurichten, das zum Beispiel von IBM unter CP/M-86 benutzt wird. Sinnvoll genutzt werden kann dies, wenn ein 5¼-Zoll-Laufwerk angeschlossen wird und ein Anwender etwa eine Textdatei, die im Büro auf einem IBM geschrieben wurde, zu Hause auf dem CPC weiterbearbeitet.

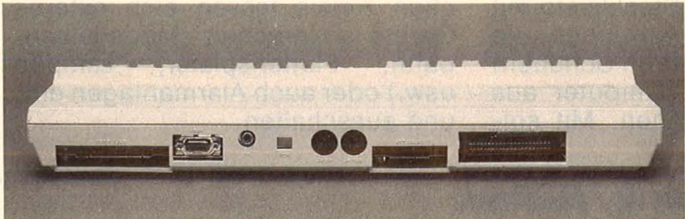
An das Expansion-Board kann auch eine RS232C-Schnittstelle angeschlossen werden, zum Beispiel wenn aus früheren Computerzeiten ein entsprechender Drucker vorhanden ist, der sich nicht an den Centronics-Ausgang anschließen läßt.

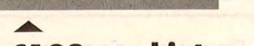
Nützlich ist die 50polige Leiste auch für Musikexperten: Von Schneider-Data wird ein Analog/Digital- bzw. Digital/Analog-Wandler namens „ADA-1" angeboten, der sich direkt an eine Stereoanlage oder andere Musiksysteme anschließen läßt und durch alle Programmiersprachen ansteuerbar ist. Weitreichende Klangmanipulationen (über den Musik-Chip AY-3-8912) in bester Qualität sind realisierbar. Wem diese Anschaffung zu kostspielig ist oder wer lediglich die normale Tonausgabe zur Stereoanlage oder einem Kopfhörer umleiten will, kann dies über die Anschlußbuchse am linken Gehäuseteil erreichen.

Commodore 64/128: Von Normung keine Spur

Für keinen anderen Computer wurden und werden so viele Zusatzgeräte angeboten wie für den Commodore 64. Da auch das Nachfolgemodell, der Personal-Computer C 128 die gleichen Schnittstellen vorweist wie der 64er (und darüber hinaus noch einige mehr besitzt), kann die bisherige Peripherie voll genutzt werden.

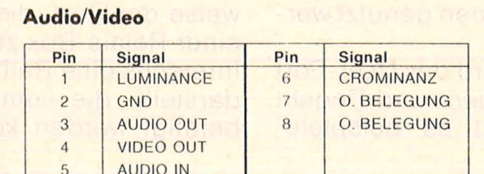
Mit Ausnahme der Joystick-Buchsen und des Fernsehantennenanschlusses ist keine der Commodore-Schnittstellen genormt. Das bereitet Usern, die auf den freien Markt ausweichen wollen, gewisse Anschlußschwierigkeiten; doch findige Fremdfirmen haben diese Probleme schon vor langem erkannt und eine nahezu unüberschaubare Fülle an Adaptern, Schnittstellenkonvertern und funktionstüchtigen Zusatzschnittstellen entwickelt, so daß es heute möglich ist, fast alles mit dem Commodore zu verbinden, was die Computerwelt hervorbringt. Trotzdem gibt es ein grundsätzliches Ärgernis beim C 128: Die nützliche und professionelle 80-Zeichen-Darstellung ist nur mit einem neunpoligen Spezialkabel möglich, das – ähnlich wie die Joystick-Anschlüsse – leicht angeschrägte Ecken vorweist. Für Umsteiger vom C64 zum 128er ist es also nicht möglich, mit dem alten Monitor 80 Zeichen darzustellen; hier stehen lediglich die gewohnten 40 Zeichen zur Verfügung, die über eine achtpolige Rundsteckerbuchse gespeist werden.



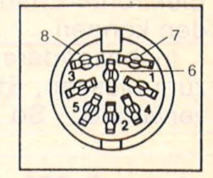


C128 von hinten: *Von links nach rechts findet man Steckmodul-Eingang, Kassettenport, Serieller Ausgang, Video-Buchse, TV-Buchse, RGBI-Buchse und User Port*

Audio/Video



Pin	Signal	Pin	Signal
1	LUMINANCE	6	CROMINANZ
2	GND	7	O. BELEGUNG
3	AUDIO OUT	8	O. BELEGUNG
4	VIDEO OUT		
5	AUDIO IN		

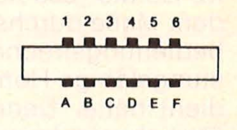


Anschluß eines Monitors: *Die achtpolige Rundsteckerbuchse ermöglicht keine 80-Zeichen-Darstellung*

Von links nach rechts: *2× Joystick, Reset, Ein/Aus und Stromversorgung*

Cassette

Pin	Signal
A-1	GND
B-2	+5V
C-3	CASSETTE MOTOR
D-4	CASSETTE READ
E-5	CASSETTE WRITE
F-6	CASSETTE SENSE



Der Kassettenport: *Die 5-Volt-Spannungsquelle ist für den Anschluß eines Fremddruckers interessant*

Auf der anderen Seite haben die beiden Schnittstellen zum Bildschirm einen Vorteil: Es ist nämlich möglich, zwei Sichtgeräte gleichzeitig mit Darstellungen zu versorgen. Dies ist für Anwender interessant, die zum Beispiel Text und Grafik gleichzeitig vor Augen haben wollen. Als Schnittstelle zum Kassettenrecorder (Datacorder) dient beim C64/128 eine Steckleiste mit sechs Doppelanschlüssen.

Auch für Anwender, die keinen Kassettenrecorder bzw. Datacorder einsetzen, ist diese Schnittstelle hochinteressant, weil der Anschluß B-2 eine 5-Volt-Spannung liefert, die für den Betrieb vieler Commodore-fremder Drucker notwendig ist. Der sechspolige serielle Rundbuchsenanschluß für den Commodore-Drucker liefert keine entsprechende Spannung. Der Anschluß eines Fremddrukkers ist jedoch für viele Anwendungen sinnvoll, weil der freie Markt eine Reihe von intelligenten Drukker-Interfaces hervorgebracht hat, die Funktionen ausführen, die normalerweise umständlich programmiert werden müßten; zum Beispiel eine Hardcopy-Funktion (Witte-Interface) oder sofortige Verfügbarkeit des hochauflösenden Grafikmodus (Rocke-Interface). Daneben ist es möglich, die verschiedenen Computer-Modi (Grafik/Großbuchstaben, Klein-/Großbuchstaben) durch einfache Parameterangaben im OPEN-Befehl original zum Drucker zu übertragen, so daß auch ein Fremdgerät (zum Beispiel Epson, Star) die Grafikzeichen des Commodore ausdrucken kann (Wiesemann-Interface).

Die Modulsteckleiste des Commodore mit zweimal 22 Anschlüssen, die ursprünglich für EPROM-Software konzipiert war (fest programmierte Software auf Speicher-Chips), hat inzwischen für andere Funktionen an Bedeutung gewonnen. Hier ist besonders der Betrieb von Akustikkopplern hervorzuheben. So wird zum Beispiel der Ascom-Akustikkoppler mit Hilfe eines intelligenten Interface am Modulsteckplatz des C64/128 angeschlossen und ist nach Laden der Diskettenbetriebs-Software ohne weiteren Programmieraufwand einsatzbereit.

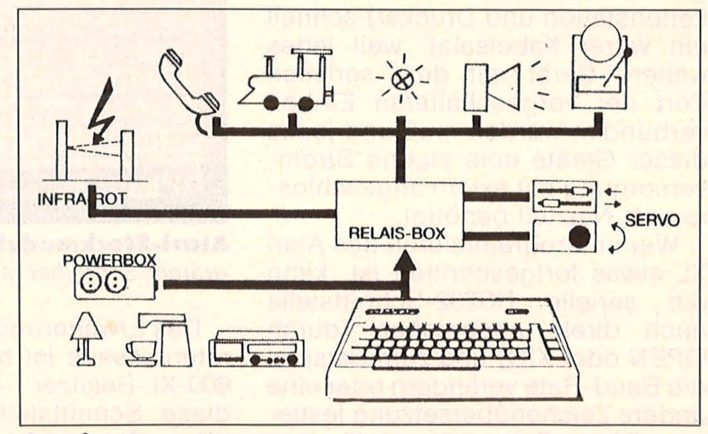
Speziell für Bastler und Computer-Experimentatoren ist die letzte Schnittstelle, das sog. User-Port, interessant. Diese Steckleiste aus zweimal zwölf Anschlüssen bietet äußerst flexible Funktionsmöglichkeiten, die allerdings mehr von

Pin	Signal	Bemerkung
1	GND	
2	+5V	MAX. 100 mA
3	RESET	
4	CNT1	
5	SP1	
6	CNT2	
7	SP2	
8	PC2	
9	SER. ATN IN	
10	9 VAC	MAX. 100 mA
11	9 VAC	MAX. 100 mA
12	GND	
A	GND	
B	FLAG2	
C	PB0	
D	PB1	
E	PB2	
F	PB3	
H	PB4	
J	PB5	
K	PB6	
L	PB7	
M	PA2	
N	GND	

1 2 3 4 5 6 7 8 9 10 11 12

A B C D E F H J K L M N

Messen, Steuern, Regeln: *Viele Erweiterungen greifen auf das User-Port zurück*



C64 im Zentrum: *Die Relais-Box dient als Grundlage zur Steuerung externer Geräte*

Elektronik-Erfahrenen genutzt werden können.

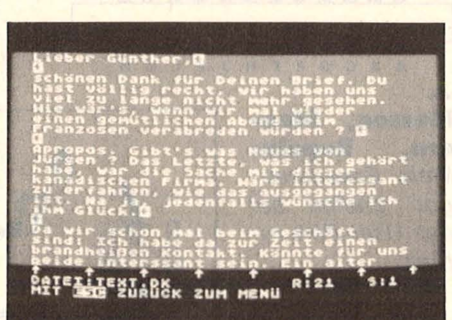
Insbesondere wird das User-Port zum Messen, Steuern und Regeln verwendet. So ist es beispielsweise denkbar, die Steckleiste mit einer Relais-Box zu verbinden, die ihrerseits eine Reihe von Schaltern darstellt, die vom Computer aus betätigt werden können. Mit solchen Relais lassen sich externe Geräte ansprechen (Modelleisenbahn, Plattenspieler, Lampen usw.) oder auch Alarmanlagen ein- und ausschalten.

Atari XL: Kabelsalat am seriellen E/A-Port

Aufgrund des großen Erfolges von Commodore und später auch Schneider konnten sich die Atari XL-Geräte (600 XL, 800 XL) nicht in dem Maße durchsetzen, wie es der bedienungsfreundliche und leistungsfähige Home-Computer verdient hätte. Dennoch hat sich – ähnlich wie beim TI 99/4A – ein hartnäckiger Kern von begeisterten Atari-Usern gebildet, die in kleinen Clubs die nur spärlich bekanntgegebenen Systemadressen und Schnittstellenbelegungen ausprobieren mußten. Der allgemeinen Unart folgend, hatte auch Atari für seine XL-Geräte keine genormten Schnittstellen vorgesehen, sondern versucht, den Anwender an die hauseigene Peripherie zu binden. Da sich der Rechner nicht so immens durchsetzte, hielt sich auch das private Schnittstellen-Engagement von Fremdfirmen gegenüber dem Atari in Grenzen. So konnte der User in vielen Fällen tatsächlich nur auf die Atari-Geräte zugreifen, was vielleicht auch ein Grund für den schleppenden Erfolg gewesen sein mag.

Wichtigste Schnittstelle ist zweifellos der serielle Ein-/Ausgabe-Port an der rechten Geräterückseite. Über diesen 13poligen Anschluß werden sämtliche Peripheriegeräte gesteuert: Kassettenrecorder, Drucker, Plotter, Diskettenstation. Allerdings ergibt sich beim Betrieb mehrerer gleichzeitig angeschlossener Geräte (etwa Diskettenstation und Drucker) schnell ein wirrer Kabelsalat, weil jedes weitere Gerät mit dem seriellen Port der vorgeschalteten Einheit verbunden werden muß und jedes dieser Geräte eine eigene Stromversorgung mit extern angeschlossenem Netzteil benötigt.

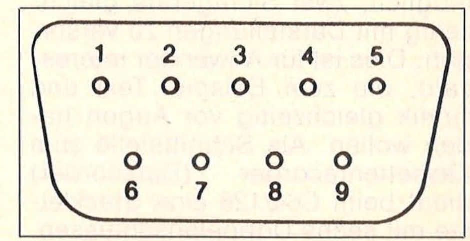
Wer im Programmieren des Atari XL etwas fortgeschritten ist, kann die serielle RS232-Schnittstelle auch direkt ansprechen (durch OPEN oder XIO) und zum Beispiel die Baud-Rate verändern oder eine andere Zeichenübersetzung festlegen. Dieser Port wird auch für spezielle Drucker-Interfaces benutzt, die atarifremde Drucker ansprechen (zum Beispiel das Star-Interface, mit dem die serielle Datenausgabe zur parallelen zusammengefaßt wird und dadurch jeder marktübliche Centronics-Schnittstellen-Drucker betrieben werden kann). Als Bildschirmschnittstellen bieten die XL-Geräte den üblichen Fernseh-Antennenstecker sowie eine Rundbuchse für Monitoranschluß. Hier ist es also möglich, zum Beispiel einen 40-Zeichen-Hantarex- oder Sanyo-Monitor anzuschließen und damit die Bildqualität erheblich zu steigern. Der Steckmodul-Eingang (Slot) an der Oberseite dient hauptsächlich für EPROM-Software, die gegenüber der Disketten- bzw. Kassetten-Software den Vorteil hat, gleich nach dem Einschalten des Gerätes ohne jede Wartezeit zur Verfügung zu stehen. Viele Computerspiele und kommerzielle Programme wurden von Atari (hauptsächlich aus Kopierschutzgründen) in Form von Steckmodulen angeboten. Populärster Vertreter ist hier das nahezu professionelle Textverarbeitungsprogramm Schreiber, das heute von Atari nicht mehr produziert wird und für den es inzwischen eine Art ,,Schwarzmarkt" gibt, auf dem das Programm zu höchstbietenden Preisen gehandelt wird.



Atari-Steckmodul: *Das Textprogramm Schreiber ist gefragt*

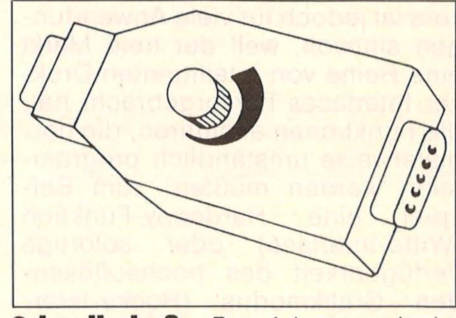
Das Erweiterungsport an der Geräterückseite ist hauptsächlich für 600-XL-Besitzer interessant. An diese Schnittstelle wird das 64-KByte-Zusatzmodul angeschlossen, wodurch der 600 XL dem 800 XL an Speicherkapazität gleichwertig wird. Leider ist diese Speichererweiterung relativ schwer, so daß sie gewissermaßen an der Rückseite heraushängt und unstabil wirkt. Eine unvorsichtige Berührung kann unter Umständen zum Absturz des Computers führen.

An der rechten Seite der XL-Geräte befinden sich schließlich noch zwei neunpolige Joystick-Schnittstellen, die das bekannte, von Atari eingeführte abgeschrägte Aussehen zeigen und in die praktisch alle handelsüblichen Joysticks passen.



Die Pinbelegung des Joystickports: *1 Joystick hoch, 2 Joystick runter, 3 Joystick links, 4 Joystick rechts, 5 Paddle links, 6 Feuerknopf, 7 +5 Volt, 8 Masse, 9 Paddle rechts*

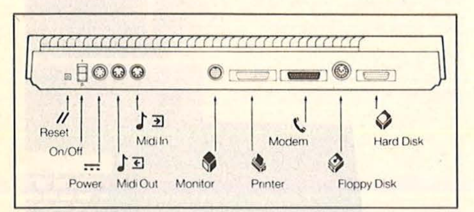
Die früheren Atari-Computer 400 und 800 besaßen übrigens noch vier solcher Joystick-Anschlüsse. Wem die Feuerkraft oder die Schußfolge der normalen Joysticks nicht genügt, kann für bestimmte Spiele zusätzlich einen Schnellfeuer-Adapter anschließen, der sich von einem Schuß pro Sekunde bis 20 Schüsse stufenweise regulieren läßt.



Schnellschuß: *Der Adapter erlaubt bis zu 20 Impulse pro Sekunde*

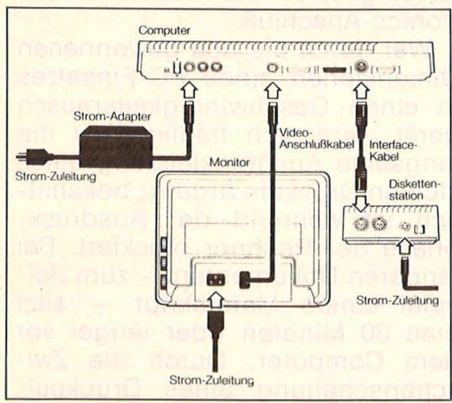
Atari ST: Über dem Durchschnitt anschlußfreudig

Mit den ST-Modellen (260, 520, 1040) ist Atari eindeutig der Sprung in die Profiklasse gelungen. Dank einer komfortablen Benutzerführung mit Hilfe der Fenstertechnik, die in das Atari-Betriebssystem TOS eingebettet ist sowie durch weitgehend genormte Schnittstellen, gelang es Atari, mit den ST-Geräten in kurzer Zeit die Gunst der Käufer zu erringen.



Vielseitig: *Die Rückfront des Atari ST*

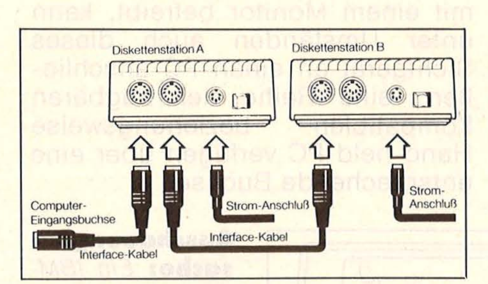
Über die üblichen Schnittstellen hinaus (Drucker, Joystick, Maus usw.) zeigen sich die ST-Computer besonders anschlußfreudig; so weist die Rückfront zwei fünfpolige Midi-Anschlüsse vor (für Midi-Ein- und -Ausgang). Midi (Musical Instruments Digital Interface) ist ein serielles Bussystem für musikelektronische Geräte. Praktisch alle Hersteller von elektronischen Musikinstrumenten (Synthesizer, Drums, Effektgeräte usw.) haben sich inzwischen diesem Standard angeschlossen, so daß in Verbindung mit einem Computer (in diesem Fall Atari ST) komplette Musikanlagen und Heimstudios zentral gesteuert werden können. Je nach Software ist es zum Beispiel möglich, Musiknoten zu speichern, simultan oder sequenziell (ständige Wiederholung einer Melodiefolge im Hintergrund) abzuspielen oder die vorhandenen Melodien rhythmisch und klanglich extrem zu manipulieren.



Kabelsalat: *Jedes Zusatzgerät benötigt eine eigene Stromversorgung*

Hinsichtlich der Stromversorgungen hat Atari auch bei den ST-Modellen die etwas unglückliche Handhabung mit externen Netzteilen beibehalten. So wird für die Konsole und jedes Diskettenlaufwerk ein eigenes Netzteil benötigt, was leicht wieder zum Kabelsalat führt. Allerdings wird für den Monitor keine eigene Stromversorgung verwendet, sondern ein entsprechendes (ungenormtes) Kabel von der 13poligen runden Monitorschnittstelle zum Bildschirm geführt, wobei Pin 8 das Sichtgerät mit der notwendigen 12-Volt-Spannung versorgt.

Das Diskettenlaufwerk wird ebenfalls über eine Rundbuchse angeschlossen, die jedoch gegenüber der Monitor-Schnittstelle 14 Leitungen aufweist, wobei Pin 14 für die Schreibschutz-Kontrolle zuständig ist. Wer den Nervenkitzel nicht scheut, kann diese Leitung ablöten und die Schreibschutz-Funktion außer Betrieb setzen.

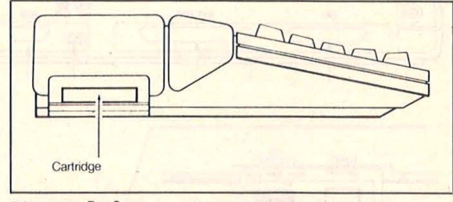


Ohne Probleme: *Der Anschluß mehrerer Diskettenlaufwerke funktioniert nach dem Kettenprinzip*

Wer sich mit einem Diskettenlaufwerk nicht zufriedengibt und ein zweites Laufwerk anschließen will, muß insgesamt fünf Schnittstellen-Verbindungen auf sich nehmen: Stromversorgung für Laufwerk A, Computer-Eingangskabel, Interface-Kabel von Laufwerk A zu Laufwerk B und Stromversorgung für Laufwerk B. Einfach wird hingegen der Anschluß einer Harddisk (Festplatten-Speichersystem) gemacht. Wo bei anderen, weitaus teureren Personal-Computern das gesamte Gerät für den Einbau einer Festplatte auseinandergenommen werden muß, bietet der ST einen einfachen Kabelanschluß über eine 19polige Steckleiste, die seitlich ähnlich abgeschrägt ist wie eine Centronics-Schnittstelle.

Anwenderfreundlich sind auch die beiden Schnittstellen für den Drucker und Modem (bzw. Akustikkoppler). Beim Druckeranschluß wurde auf das bewährte 8-bit-Centronics-System zurückgegriffen, wodurch eine lebendige Vielfalt an frei angebotenen Druckern mit dem Atari ST betrieben werden kann. In der Regel muß dazu lediglich der gewünschte Zeichensatz installiert werden. Sofern der eingesetzte Drucker zur Darstellung von Bit-Image-Grafik (Einzelpunktgrafik) eingerichtet ist, läßt sich jederzeit ein bildschirmidentischer Ausdruck (Hardcopy) aufrufen.

Rechts neben der Drucker-Schnittstelle (von hinten gesehen) befindet sich eine marktübliche RS232C-Schnittstelle, die zur Aufnahme eines Modem- oder Akustikkoppler-Anschlusses dient. Diese Schnittstelle läßt sich sehr bequem vom Betriebssystem aus für den gewünschten Einsatz konfigurieren. Man muß dazu die Option „RS232 Konfiguration" anwählen. Es erscheint daraufhin ein Fenster mit allen wichtigen Einstellmöglichkeiten (Baud-Rate, Paritäts-Prüfung, XON/XOFF-Protokoll usw.). In gleicher Weise läßt sich auch der Druckeranschluß bezüglich Papierart, Zeilenabstand und ähnliches einstellen.



Nostalgie: *Zur Aufnahme von EPROM-Software ist wie bei der XL/XE-Serie ein Schacht vorgesehen*

An der linken Geräteseite befindet sich ein Cartridge-Schacht, der die gleiche Funktion übernimmt wie der Slot der XL-Modelle: er dient zur Aufnahme von EPROM-Software und besitzt insgesamt 40 Datenleitungen, die von den einsetzbaren Cartridges unterschiedlich genutzt werden. Bastler können diese Schnittstelle auch experimentell einsetzen, da zum Beispiel die Pins 1 und 2 je 5 Volt

Stromspannung liefern, die sich zur Versorgung externer Schaltkreise (Steuerungselektronik) heranziehen lassen. Besondere Wichtigkeit erhalten die beiden neunpoligen Buchsen an der rechten Geräteseite; die erste stellt die Verbindung zum wichtigsten Steuerinstrument (der Maus) her, an der zweiten läßt sich ein normaler Joystick anschließen. Ein Joystick-Anschluß ist überdies auch mit Buchse 1 möglich. Beim 1040 STF befinden sich diese Schnittstellen an der Geräteunterseite.

IBM PC und Kompatible: Schnittstellen für Umsteiger

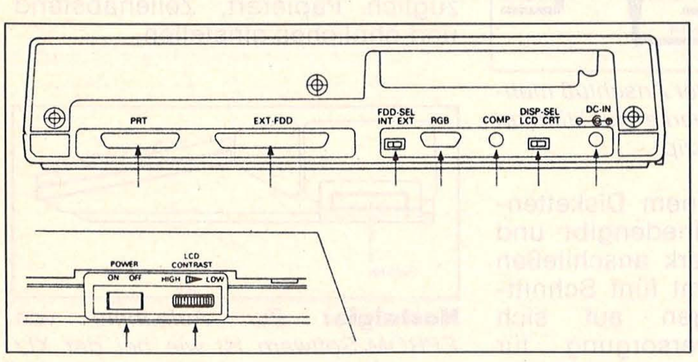
Durch rasant fallende Preise und wachsendes Qualitätsniveau ist auch für private und halbprofessionelle Anwender der Kauf eines regulären (das heißt IBM-kompatiblen) Personal Computers durchaus ein diskussionswürdiges Thema. Es gibt heute Angebote, die ebenso günstig sind wie etwa eine komplette Anlage des Commodore 128 mit Drucker, Monitor und Laufwerk.

Der Vorteil, den ein IBM oder ein Kompatibler bietet, liegt vor allem in den einheitlichen Anschlußmöglichkeiten für Peripherie sowie in dem enormen Angebot an austauschbarer, lauffähiger Software.

Im allgemeinen muß sich ein PC-Besitzer keinerlei Gedanken um die Schnittstellen, deren Konfiguration und Betriebsleistungen kümmern, weil dies normalerweise durch die eingesetzte Profi-Software übernommen wird. Aus diesem Grund gehen die Handbücher der Geräte auch kaum auf die Anschlüsse der Geräte ein. Dank des ausgereiften MS-DOS-Betriebssystems können viele Schnittstellen-Zugriffe im Direktmodus auch ohne jede Programmierkenntnisse durchgeführt werden; so ist es zum Beispiel möglich, sämtliche Bildschirmausgaben durch den MS-DOS-Systemaufruf PRN zum Drukker umzuleiten. Ähnlich einfach – nämlich durch Drücken von CONTROL und PrtSC – erhält man einen bildschirmidentischen Textausdruck (Hardcopy).

Interessant werden die Schnittstellen eines PC für Leute, die von Home-Computer- auf die PC-Ebene umsteigen wollen. In diesen Fällen sollte man sich ein Gerät aussuchen, das den Betrieb von möglichst vielen, bereits vorhandenen Peripheriegeräten ermöglicht. Denn ein Umsteiger will schließlich nicht unbedingt seinen Monitor oder Drucker zum Fenster rauswerfen, nur weil der Kompatible einen anderen Zeichenvorrat ausdruckt oder einen speziellen Bildschirmanschluß bietet.

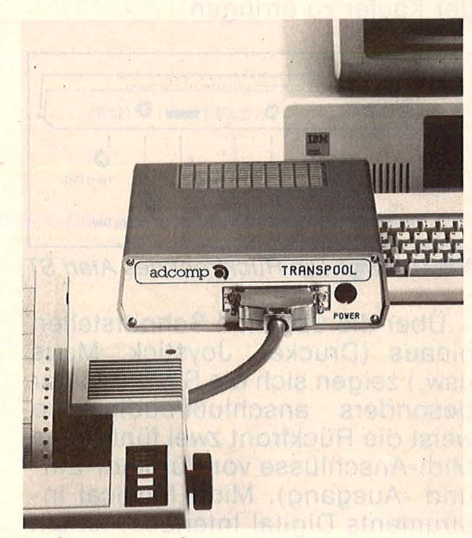
Der Toshiba T1100 zum Beispiel verfügt über eine RGB-Monitor-Schnittstelle, die identisch ist mit der des Commodore 128. Ein Umsteiger könnte in diesem Fall also den vorhandenen 80-Zeichen-Bildschirm anschließen. Wer zufällig einen anderen Home-Computer besitzt und dieses Gerät über einen einfachen Composite-Stecker mit einem Monitor betreibt, kann unter Umständen auch dieses Sichtgerät an einen PC anschließen; eine Reihe der tragbaren Kompatiblen beziehungsweise Hand-held-PC verfügen über eine entsprechende Buchse.



Anschauungssache: *Ein IBM-kompatibler Hand-held-Computer von hinten und von links*

Für den Geldbeutel ist neben diesen Hardware- beziehungsweise Peripherie-Voraussetzungen auch systemtreibende Software von Nutzen. So gibt es ein Utility (Dienstprogramm) namens Printklick, das vor dem Beginn der Arbeit mit einer Anwender-Software geladen wird und im Hintergrund resistent bleibt (das heißt, es wird auf Tastendruck aufgerufen oder abgeschaltet, ohne daß das Hauptprogramm gelöscht wird).

Mit Printklick ist es unter anderem möglich, einen einfachen – in der Home-Computer-Szene häufig eingesetzten – Epson FX-80-Drucker IBM-kompatibel zu machen. Das bedeutet, der FX-80 verfügt dann über den vollen IBM-Grafikzeichenvorrat, ohne seine typischen Eigenschaften zu verlieren. Möglich wird dies durch die Software-gesteuerte Programmierung von Download-Characters (benutzerspezifischer Zeichensatz). Somit können auch alle übrigen Drucker, die über einen Epson-identischen Befehlssatz verfügen, durch Printklick IBM-kompatibel gemacht werden. Erforderlich ist lediglich ein marktüblicher Centronics-Anschluß.

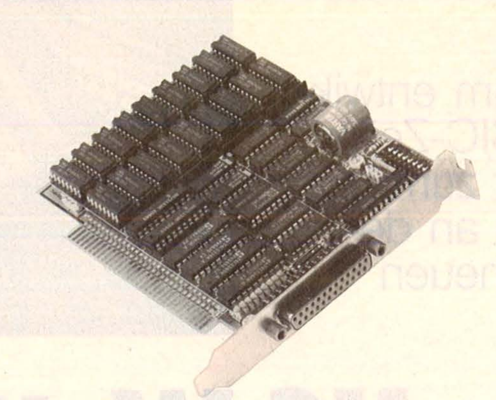


Zeitersparnis: *Ein Drucker-Spooler für IBM mit großer Speicherleistung*

Wer durch die neu gewonnenen Dimensionen eines PC-Einsatzes in einen Geschwindigkeitsrausch gerät, wird sich häufig über die langsame Ausdruckleistung eines kleinen Druckers ärgern; bekanntlich ist während der Ausdruckphase der Rechner blockiert. Bei längeren Dokumenten – zum Beispiel einem Manuskript – sitzt man 30 Minuten oder länger vor dem Computer. Durch die Zwischenschaltung eines Druckpuffers (auch Spooler oder Pufferspeicher) kann der Datentransfer an

der computereigenen Schnittstelle entlastet werden, weil alle auszudruckenden Zeichen in einem Puffer zwischengelagert und Stück für Stück vom Drucker abgerufen werden. Während der Ausdruck noch läuft, ist der Computer wieder betriebsbereit.

Die ultimative Lösung für Schnittstellen-Probleme bieten Geräte, die serielle und parallele Datenübertragung interaktiv „unter einem Dach" verwalten können. Meist sind diese intelligenten Interfaces mit einem Pufferspeicher kombiniert (zum Beispiel der Gerb-Spooler DP 100), so daß neben der Möglichkeit, Daten aus serieller Schnittstelle in eine parallele Schnittstelle umzuleiten (und umgekehrt) auch ein externer Speicher für auszudruckende oder übertragbare Daten zur Verfügung steht. Allerdings sind diese Geräte nicht ganz billig.



Komplettlösung: *Eine RAM-Erweiterungskarte mit Druckeranschluß*

Anstelle der externen Schnittstellen-Peripherie erlauben die IBM- und kompatiblen PC auch den Einbau von Erweiterungskarten, die einerseits den Arbeitsspeicher des Computers vergrößern und andererseits verschiedene Zusatz-Schnittstellen bieten; hierzu zählen ein zweiter Druckeranschluß, ein Modem-Anschluß (RS232C-Schnittstelle) oder ein Maus-Anschluß, der sich auch für Joysticks eignet.

Alfred Görgens

Sinclair Spectrum: Zusatz-Interfaces gefragt

Interessant wird die Arbeit mit dem Sinclair Spectrum erst, wenn man die entsprechenden Erweiterungs-Schnittstellen besitzt. Allem voran das Interface I von Vobis, mit dem der Anschluß der Microdrives möglich wird. Für rund 200 Mark bietet es zudem die Möglichkeit, mehrere Rechner zu verbinden. Das V.24-Interface von Müller in Flensburg enthält neben der erwähnten seriellen Schnittstelle ein erwähnenswertes Terminalprogramm.

Für rund 1100 Mark gibt es bei verschiedenen Herstellern das Diskettenlaufwerk Discovery von Opus. Das Interface I und die Discovery haben eines gemeinsam: Sie vermeiden den „Schnittstellensalat", sind leicht anzusteuern und liefern außerdem einen schnellen Massenspeicher mit.

Die Discovery-Schnittstelle kann (bisher) jedoch außer für die Druckeransteuerung nur für DFÜ eingesetzt werden, wenn man das entsprechende Programm kauft. Für andere Zwecke ist sie derzeit nicht einzusetzen. Der hohe Preis – dafür bekommt man drei Computer – ist allerdings etwas abschreckend. Das Kempston-Joystick-Interface als „Zugabe" kann als weiterer Pluspunkt gewertet werden. Das Microdrive und die Floppy weisen in der Ansteuerung kaum Unterschiede auf, weshalb es auch nur wenige Programm-Übertragungsprobleme gibt.

Das Wafadrive machte zunächst dem Interface I Konkurrenz, weil es zwei Laufwerke bietet. Gegen dieses Gerät sprechen inzwischen die Erfahrungen. Die Ansteuerung ist schwierig, die Probleme der Software-Kompatibilität noch lange nicht alle gelöst. Der englischen Dokumentation sind kaum Hilfen zu entnehmen. Dieses Gerät wird von Programmherstellern kaum unterstützt, weil es nicht sehr verbreitet ist. Vermutlich gibt es auch deshalb noch keine Hardcopy- oder DFÜ-Software, die hiermit läuft. Nicht einmal der Druckeranschluß ist problemlos durchzuführen. Privatlösungen sind allerdings möglicherweise über diverse Sinclair-Clubs erhältlich.

Die Centronics-Schnittstellen von Dorsch und Microcomputerladen machen bei der Druckersteuerung kaum Kopfzerbrechen. Beim Dorsch-Interface ist das Kabel fest angeschlossen und braucht nur noch beim Drucker eingesteckt zu werden. Allerdings läßt sich die Code-Umwandlung der Zeichen über ASCII 127 nicht abschalten.

Vielseitig: *Die Schnittstellenauswahl beim Sinclair Spectrum in Verbindung mit dem Interface I*

Von den Joystick-Interfaces sind eigentlich nur die beiden genannten interessant. In beiden Fällen ist die Benutzung und auch der Anschluß ganz einfach durchzuführen. Das Kempston-Interface verträgt sich allerdings wegen verschiedener Abfragen nicht mit anderen als dem Original-ROM des Spectrum. Das Interface II hat den Vorteil, daß es Tastatur-Eingaben simuliert (also diese Software auch ohne Interface II funktioniert). Der Programm-ROM-Steckplatz dürfte in Anbetracht der Ladezeiten bei Microdrives nicht so interessant sein; er wird auch wenig unterstützt.

Empfehlenswert zur Vermeidung größerer Probleme sind folgende Kombinationen:
- Interface I, Microdrive, Interface II, serielles Interface am Drucker oder
- Discovery, Centronics-Interface am Drucker, evtl. V.24-Interface von R. Müller.

In beiden Fällen sind alle notwendigen Elemente vorhanden; die Kombinationen vertragen sich mit fast aller Software (werden auch unterstützt). Sie sind gut bis besser dokumentiert (Schnittstellen-Belegung, Programmierhinweise, Programmanpassung) und gewährleisten eine zufriedenstellende Datensicherheit.

Erika Hölscher



Wenn Sie ein Programm entwikkeln, das die C64-BASIC-Zeilen auf den kleinen Bruder zum Laufen bringt, nehmen Sie an der Verlosung eines nagelneuen Commodore 64 teil

Großer HC-Wettbewerb für den Wühltischrechner

Als Nachfolger für den „Volkscomputer VC-20" war er entwickelt worden, der C16 und sein kleiner Radiergummibruder C116. Doch statt zu einem Verkaufsschlager zu werden wie sein Vorgänger, diente er in den ausgedehnten Frankfurter Lagerhallen von Commodore als Staubfänger: Die einen potentiellen Käufer warteten lieber auf den Amiga, die anderen stiegen um auf Konkurrenzrechner, die zu fast dem gleichen Preis das vier- bis achtfache an Speicherplatz boten. Als dann bei Comodore das Geld knapp wurde, besann man sich in der Vorstandsetage und verschleuderte Zehntausende von C16 an Aldi und andere Kaufhausketten. Und deshalb freut sich Oma Piepenbrink bei ihrem allsamstäglichen Einkauf, daß diese Computer, von denen ihre Enkelin immer schwärmt, schon für unter hundert Mark zu haben sind. Und kauft einen für den nächsten Geburtstag.

Doch spätestens am zweiten Tag fragt sich die so Beschenkte: „Was soll ich mit dem Ding?" Während die Klassenkameraden, die zu Weihnachten einen C64 geschenkt bekamen, Space-Invaders bis zum Umfallen spielen können, scheitert das beim C16 daran, daß es kaum solcherlei Software dafür gibt. Vergleichsweise vernünftigere Anwendungen, Datenfernübertragung oder die Steuerung der Modelleisenbahn via Computer scheitern daran, daß der C16 an der Stelle, wo VC20 und C64 ihren Userport haben, ein Loch hat – für den Kassettenrecorder.

Bleibt also nur, selber zu programmieren oder nach Studium des Handbuchs festzustellen, daß nicht nur das BASIC des C16 kompatibel zu dem des C64 ist, sondern auch das Diskettenlaufwerk VC1541. Dem Grundsatz des Hauses Commodore – ein neuer Computer hat zu allen anderen inkompatibel zu sein – trägt aber wenigstens die Adressbelegung Rechnung. 64er Maschinenprogramme laufen sowieso nicht und trifft der kleine schwarze Computer in BASIC-Programmen seines großen Bruders auf PEEKs und POKEs, dann tut er bestenfalls irgend etwas Unerwartetes, zumeist aber etwas Erwartetes: Er stürzt ab. Um Programme lauffähig zu kriegen, helfen also nur etliche Nachtschichten, um C64-Zeilen in die entsprechenden C16-Zeilen zu verwandeln. Da Computerbesitzer bekanntlicherweise sehr faul sind, bietet sich für diese stupide Arbeit die Überlegung an, diese mühevolle Konvertierungsarbeit den Computer machen zu lassen.

Faulheit muß man unterstützen. Und darum stiften wir demjenigen Leser, der das beste Konvertierungsprogramm von C64 auf C16 schreibt, einen nagelneuen C64.

Das geforderte Programm muß folgendes leisten: Es muß BASIC-Programme, die für den C64 geschrieben worden sind, selbständig einlesen, PEEK- und POKE-Befehle erkennen und anhand der abgedruckten Tabelle ersetzen. Wird in dem C64-Programm auf den Sound- oder Grafikchip zugegriffen (beim C64 sind das die Adressen ab 53248), dann soll es sich die Zeilennummern merken und am Schluß der Konvertierung den Besitzer auf diese Zeilennummern hinweisen.

Diese Befehle müssen dann von Hand mittels des wesentlich mächtigeren BASICS des C16 (BASIC 3,5 statt BASIC 2,0), das die Befehle BOX, DRAW, CIRCLE, SOUND oder VOL kennt, ersetzt werden. (Oder bei Sprites, die der C16 nicht kennt, rausgeworfen werden.)

Da bei Kassettenspeicherung beide Geräte zwar das gleiche Aufzeichnungsformat, aber unterschiedliche Taktfrequenzen benutzen, ist es nur möglich, vom C64 auf Diskette geschriebene Programme mit dem C16 zu lesen und zu konvertieren.

Es gibt verschiedene Lösungsansätze:

1. Eine Möglichkeit wäre es, das Programm noch direkt auf der Diskette umzuwandeln. Dazu benötigt man einen speziellen Diskmonitor, der die Tracks und Sektoren der Diskette, auf der sich das zu konvertierende File befindet, nach den Tokens (Kürzeln) der Befehle PEEK und POKE durchsucht, die dazugehörigen Adressen herausfindet und ersetzt. Der eingebaute Maschinensprache-Monitor TEDMON des C16 könnte dazu behilflich sein.
2. Wem das zu kompliziert erscheint, der kann sich auch die Möglichkeit des C64 und des C16 zunutze machen, Programmlistings in sequentielle Dateien umzuwandeln:

OPEN 8,8,8,"SEQUENZ,S,W" :CMD8: LIST:PRINT#8:CLOSE8

schreibt das geladene Programm in einen sequentiellen File namens „Sequenz". Das in HC 6/86 wandelt diese sequentiellen Files wieder in ein Programm zurück, indem

es zeilenweise das Listing auf den Bildschirm druckt, das Programm anhält und über den Tastaturpuffer genau auf der Zeile ein ‹RETURN› auslöst, womit die Zeile in den Programmspeicher übernommen worden ist. Auf gleiche Weise löscht sich das Konvertierungsprogramm anschließend aus dem Speicher, (Zeilennummern drucken, in denen das Programm steht, Cursor drauf und ‹RETURN›) so daß nur noch das zu konvertierende Programm im Speicher ist, das mit SAVE abgespeichert werden kann. Zwischen dem Einladen des sequentiellen Files und der Übernahme der Zeile mit ‹RETURN› müßte bei einem Umwandlungsprogramm C64-in-C16 ein Algorithmus stehen, der PEEKs und POKEs erkennt und den Programmzeilenstring entsprechend der Tabelle umändert.

Auch wenn dieses Programm vorwiegend C16-Benutzern zugute kommt, sind auch alle Leser, die einen C64 haben, aufgefordert, ihre Programmierkenntnisse unter Beweis zu stellen. *Joachim Graf*

Umwandlung der POKE-Adressen von C64 auf C16

Adresse C64	Adresse C16/116	Funktion
3-4	754-755	Vektor: Umwandlung Fließkomma/Ganze Zahl
5-6	756-757	Vektor: Umwandlung Ganze Zahl/Fließkomma
7	7	Suchzeichen/Trennzeichen
8	8	Suchzeichen/Trennzeichen, Flag für Anführungszeichen am Ende eines Strings
9	9	Cursorspalte ab letztem TAB
10	10	Flag: LOAD (=0) und VERIFY (=I) bei BASIC
11	11	Eingabepufferzeiger, Anzahl der Elemente (DIM)
12	12	Flag: Standardwert von DIM
13	13	Flag: Datentyp: 255=String 0=Numerisch
14	14	Flag: Art der numerischen Variablen (128=Ganze Zahl, 0=Fließkomma)
15	15	Flag: DATA, LIST u. ,,Garbage Collection"
16	16	Flag: Benutzerfunktionsaufruf
17	17	Flag: 0=INPUT, 64=GET, 152=READ
18	18	Flag: Vorzeichen bei trigonometrischen Funktionen, für Gleichheit bei Vergleichsoperationen
19	19	Flag: Aktuelle L/0-Kanal (CDM) Nummer, Input-Prompt (=Fragezeichen)
20-21	20-21	Ganzzahlige Werte (Adressen), Zeilennummern
22	22	Zeiger auf temporären Stringstapel
23-24	23-24	Zeiger auf letzte Stringadresse
25-33	25-33	Stapel f. temporäre Strings (Stringstack)
34-35	34-35	Suchzeiger, Übertragungszeiger, Variablenzeiger, Sprungvektor, Lesen von Zeilennummern, Zeiger für Speicherverschiebungen.
36-37	36-37	Zeiger für indirekte Speicherung
38-41	38-41	Fließkommaergebnis der Multiplikation (Mantisse)
43-44	43-44	Zeiger: Anfang BASIC-Test
45-46	45-46	Zeiger: Anfang der nichtindizierten Variablen VARTAB-1=Ende BASIC-Text)
47-48	47-48	Zeiger: Anfang der Feld-Variablen
49-50	49-50	Zeiger: Ende Feldvariablen +1
51-52	51-52	Zeiger: Anfang der Stringspeicherung
53-54	53-54	Zeiger: Stringübertragungen
55-56	55-56	Zeiger: Ende BASIC-Speicher +1
57-58	57-58	Zeiger: Derzeitige BASIC-Zeilennummer
59-60	601-602	Zeiger: Vorherige Zeilennummer (für CONT)
61-62	603-604	CHRGET-Zeiger für CONT
63-64	64-64	Zeiger: Derzeitige DATA-Zeilennummer
65-66	65-66	Zeiger: Aktuelle DATA-Adresse
67-68	67-68	Vektor: INPUT-Routine
69-70	69-70	Derzeitiger BASIC-Variablenname
71-72	71-72	Zeiger: Aktuelle Variable
73-74	74-74	Variablenzeiger für FOR/NEXT
75-96	75-96	Zwischenspeicher für BASIC-Zeiger/Daten
97	97	FAC-Exponent
98-101	98-101	FAC-Mantisse
102	102	FAC-Vorzeichen
103	103	Polynomauswertung
104	104	FAC-Überlauf
105	105	ARG-Exponent
106-109	106-109	ARG-Mantisse
110	110	ARG-Vorzeichen
111	111	Ergebnis des Vorzeichenvergleiches FAC/ARG
112	112	FAC-Niederwertigste Stelle (Rundung)
113-114	113-114	Zeiger: Kassettenpuffer
115-138	1139-144	CHRGET-Routine
121	1145-1156	CHRGOT-Routine
122-123	59-60	Zeiger: Derzeitiges Byte des BASIC-Textes
136-143	1283-1287	Eingangswert der RND-Funktion
144	144	I/O-Statuswert:ST
145	145	Flag: STOP-Taste/RVS-Taste
147	147	Flag: 0=LOAD, 1=VERIFY (Beim C16 ist das Label mit VERFCK benannt.)
148	148	Flag: ,,Zeichen gepuffert" im seriellen Bus
149	149	Zeichen im Puffer für seriellen Bus
152	151	Anzahl der offenen Dateien, Dateitabellen-Index
153	152	Standard-Eingabegerät (0=Tastatur)
154	153	Standard-Ausgabegerät (CMD) (3=Bildschirm)
157	154	Flag: 128=Direktmodus 0=Programm
160-162	163-165	Echtzeituhr (jiffy clock) ca. 1/60 sek
183	171	Länge des aktuellen Dateinamens
184	172	Logische Dateinummer
185	173	Aktuelle Sekundäradresse
186	174	Aktuelle Gerätenummer
187-188	176-178	Zeiger: Aktueller Dateiname
193-194	178-179	I/O-Startadresse
195-196	180-181	Zeiger auf Kernal-Vektoren
197	2038	Vorher gedrückte Taste
198	239	Anzahl der Zeichen im Tastaturpuffer
199	194	RVS-Flag (1=RVS 0=Normal)
200	195	Ende der logischen Zeile für Eingabe
203	198	Letzte gedrückte Taste
208	199	Flag: INPUT oder GET über Tastatur
209-210	200-201	Anfangsadresse der aktuellen Bildschirmzeile
211	202	Derzeitige Cursor-Spalte
212	203	Flag: Anführungszeichenmodus (1=ja, 0=nein)
213	204	Physische Bildschirmzeilenlänge
214	205	Derzeitige Cursor-Zeile
215	206	Temporärer Datenbereich
216	207	Flag: Einfügemodus (Inhalt=Anzahl der Einfügungen)
243-244	234-235	Zeiger: Derzeitiger Farb-RAM des Bildschirmes
245-246	236-237	Vektor: Tastatur-Dekodiertabelle
256-511	291-511	Stapelspeicher des Prozessors
512-600	512-600	BASIC-Eingabepuffer
601-610	1289-1296	Tabelle der aktiven logischen Dateinummern
611-620	1299-1306	Tabelle der Gerätenummern für jede aktive Datei
621-630	1306-1318	Tabelle der Sekundäradressen jeder aktiven Datei
631-640	1319-1328	Tastaturpuffer
641-642	1329-1330	Startadresse des RAM für Betriebssystem
643-644	1331-1332	Zeiger: RAM-Ende für Betriebssystem
645	1333	Flag: Zeitüberschreitung auf IEEE-Bus
646	1339	Derzeitige Zeichenfarbe
648	1342	Anfang des Bildschirmspeichers (Page)
649	1343	Größe des Tastaturpuffers
650	1344	Flag: Tastenwiederholung
651	1345	Zählgeschwindigkeit für Wiederholen
652	1346	Wiederholungsverzögerung
653	1347	Abfrage, ob gedrückt: SHIFT, CTRL und Commodore
654	1348	Kopie von 653 zur Abfrage, ob diese Tasten schon bei vorheriger Abfrage gedrückt waren
655-656	1349-1350	Vektor: Tastaturabfrage
657	1351	Flag für Blockierung der Umschaltung Text/Grafik mit CHR$8/9 128-SHIFT unwirksame 0=wirksam
658	1352	Flag: Automatisches Scrollen abwärts. 0=Ein ≠ 0=Aus
768-769	768-769	Vektor: BASIC-Fehlermeldung anzeigen
770-771	770-771	Vektor: BASIC-Warmstart
772-773	772-773	Vektor: BASIC-Text in Token umwandeln
774-775	774-775	Vektor: Token in BASIC-Text umwandeln (Chart-List)
776-777	776-777	Vektor: BASIC-Befehl ausführen (Character Dispatch) IEVAL
778-779	778-779	Vektor: Token-Auswertung (FRMEVEL)
780	2034	Übergabewert .A-Register
781	2035	Übergabewert .X-Register
782	2036	Übergabewert .Y-Register
783	2037	Übergabewert :Statusregister
785	1280	URS-Sprung
785-786	1281-1282	USR-Adresse
788-789	788-789	IRQ RAM-Vektor
790-791	790-791	BRK-Vektor
794-795	792-793	Kernal OPEN-Vektor
798-797	794-795	Kernal CLOSE-Vektor
796-799	796-797	Zeigt auf: Kernal CHKIN-Vektor
800-801	798-799	Kernal CHKOUT-Vektor
802-803	800-801	Kernal CLRCHN-Vektor
804-805	802-803	Kernal CHRIN-Vektor
805-807	804-805	Kernal CHROUT-Vektor
808-809	806-807	Kernal STOP-Vektor
810-811	806-809	Kernal GETIN-Vektor
812-813	810-811	Kernal CLALL-Routine
814-815	821-813	Vektor: Benutzer-IRQ
816-817	814-815	Zeigt auf: Kernal LOAD-Vektor
818-819	816-817	Kernal SAVE-Vektor
828-1019	819-1010	Kassettenpuffer
1024-2047	3072-4096	Video-RAM (Beim C16 ist das Label TEDSCN benannt)
2048-	4096-	Beginn des BASIC-Speichers
53265	65286	Grafikmodus
53270	65287	Smooth Scolling
53273	65289	Interruptregister
53274	65289	Maskenregister
53280	65305	Rahmenfarbe
53281	65301	Hintergrundfarbe 0
53282	65302	Hintergrundfarbe 1
53283	65303	Hintergrundfarbe 2
53284	65304	Hintergrundfarbe 3

Computer aus zweiter Hand

Die rapiden Preissenkungen vergangener Jahre haben ihre Auswirkung auch auf das Gebraucht-Computergeschäft. Nur mit entsprechendem Zubehör sind Käufer zu finden

Das Problem tauchte schon vor fünf Jahren auf: Da machte der TI-99/4A durch seinen Preissturz Geschichte. Von anfangs knapp 3000 Mark rutschte der Preis über 1500 Mark innerhalb kürzester Zeit auf unter 1000 Mark. Am Ende war der Rechner im Ausverkauf gar für 150 Mark zu haben. Keine Chance also für einen, der frühzeitig den Rechner gekauft hat und ihn dann aber wieder loshaben will. Ähnlichen Frust erlebten auch die ersten C64-Käufer, die vor drei Jahren immerhin satte 1500 Mark für das Gerät hinblättern mußten. Sie waren gezwungen, nachdem der Preis um über die Hälfte gefallen war, einen erheblichen Verlust einzustecken.

Die Liste läßt sich beliebig fortsetzen. Und damit wird ein Trend auf dem Gebraucht-Computermarkt deutlich: Nur noch preisstabile Rechner bringen die Garantie, daß ein großer Teil der Investitionen wieder herausspringt. So etwa der IBM PC, der über die Jahre seinen Preis nahezu konstant hielt. Einzige Chance, die sich dem Verkäufer anderer Computer bietet: Er muß durch die Beigabe von Software und Peripherie seinen Rechner für den Käufer so attraktiv wie möglich machen.

Denn Kaufinteressenten gibt es genug. Doch die haben das Glück, sich in einem übersättigten Markt in Ruhe umsehen zu können, um nach dem günstigsten Angebot Ausschau zu halten. Und bei dem Überangebot von Secondhand-Rechnern bietet sich für den Käufer auch noch die Chance, durch Handeln den Preis zu drücken. So müssen viele Anbieter mit geschmälertem Erlös als eingeplant vorliebnehmen.

In unserer Gebraucht-Computerliste führen wir die gängigsten Modelle. Die teilweise erheblichen Preisschwankungen stammen von der unterschiedlichen Ausstattung der Computer beziehungsweise der Beigabe von Zubehör. Neben dem aktuellen Stand haben wir die Gebrauchtpreise der letzten Jahre aufgeführt.

Trotz des eindeutig für den Käufer tendierenden Gebraucht-Computermarktes sollten Sie sich nicht entmutigen lassen, ein nicht mehr benötigtes Gerät anzubieten, wenn Sie das Gefühl haben, daß ein anderer noch etwas damit anfangen kann. *-wt*

Wichtige Hinweise für Verkäufer

- Am gefragtesten sind zur Zeit IBM-kompatible Personal-Computer sowie tragbare Rechner für dem Heimgebrauch.
- „Exoten" lassen sich momentan schlecht verkaufen.
- Von „Privat zu Privat" findet man am besten über den Kleinanzeigenteil von Computerzeitschriften einen Käufer.
- In verschiedenen Städten gibt es professionelle Gebraucht-Computergeschäfte. Dort kann man auch Informationen über den Gebrauchtwert einholen.
- In vielen Großstädten finden bereits Computer-Flohmärkte statt. Hier bietet sich eine ideale Möglichkeit, Käufer zu finden.
- Viele Verkäufer gehen dazu über, Gebrauchtgeräte über Mailboxen anzubieten.
- Passen Sie Ihren Gebrauchtpreis dem aktuellen Marktpreis an.
- Als Richtlinie für einen etwa zwei Jahre alten Rechner gilt etwa 60 Prozent vom Neuwert.
- Wenn Sie über Software oder interessante Peripherie verfügen, sollten Sie sich überlegen, ob Sie den Rechner zusammen mit dem Zubehör als Komplettangebot anbieten.

Die wichtigsten Tips für Käufer

- Komplettangebote sind besser als die Komponenten einzeln zu kaufen, weil in der Regel Geld gespart wird und die Geräte aufeinander angepaßt sind.
- Bei Preisvergleichen unbedingt auf das mitangebotene Zubehör (Software, Peripherie) achten.
- Vorsicht ist geboten, wenn an dem Computer herumgebastelt worden ist. Lassen Sie sich am besten bestätigen, daß sich der Computer im Originalzustand befindet.
- Aus der Kaufquittung kann man das Alter des Rechners herauslesen.
- Vor dem Kauf unbedingt prüfen: Bewegliche Teile wie Tastatur, Laufwerk, Recorder, Drucker etc. Beim Bildschirm dürfen die Zeilen auch bei längerem Betrieb nicht zu zittern beginnen.
- Machen Sie sich die Mühe, vorher den Markt und die Preise zu studieren.
- Es gibt genügend Verkäufer, die sich zum Handeln über den Preis einlassen.
- Überprüfen Sie, ob die angebotene Peripherie auch zum Rechner paßt.
- Achten Sie darauf, daß keine Kabel oder Handbücher fehlen.
- Wichtig ist die Anzahl und Funktion der Schnittstellen.
- Es gibt professionelle Händler, die auch eine Garantie für gebrauchte Geräte übernehmen.

Hersteller	Modell	Neupreis in diesem Jahr ca. in Mark	Gebrauchtpreislage in Mark			Ausstattung und Zubehör
			1986	1985	1984	
Apple	Apple II+	–	750 bis 3300	1500 bis 3200	3000 bis 4450	48 oder 64 KByte RAM, Monitor, Laufwerke, Drucker, Software
	Apple IIe	2800	1995 bis 3100	3200 bis 4800	5500	128 KB, 2 Laufwerke, Monitor, diverse Literatur und Software
	Apple IIc	2000	1500 bis 4700	2300 bis 2500	–	128 KB, Monitor, Laufwerke, Imagewriter, Software
Atari	800 XL	200	400 bis 800	400 bis 900	–	64 KB, Laufwerk, Drucker, Software
	130 XE	400	520 bis 1200	–	–	128 KB, Laufwerk, Drucker, Software
	520 ST	2700	2000 bis 2500	–	–	512 KB
Colour Genie	Colour Genie	400	400 bis 1300	350 bis 800	350 bis 500	32 KB, Monitor, Kassettenrecorder, Software
Commodore	C64	500	600 bis 1200	500 bis 1400	600 bis 1900	Kassettenrecorder, Drucker, Disketten, diverse Software
Hewlett Packard	HP-41C	690	330 bis 950	300	360 bis 2500	Drucker, diverse Peripherie, Software, Literatur
	HP-41CV	530	600	470 bis 480	380 bis 2000	Diverses Zubehör
	HP-41CX	840	400 bis 590	850	950	RAM-Erweiterung
	HP-71B	1560	850 bis 1200	1100 bis 1350	–	Diverses Zubehör
	HP-75C	3400	1250 bis 1650	1200 bis 1900	1690 bis 2750	Diverses Zubehör
IBM	IBM PC	6270	4000 bis 6800	3990	–	256 KB, Monitor, Tastatur
Schneider	CPC 464	700	500 bis 790	750 bis 800	–	Monochrom-Monitor
	CPC 464	1140	1500	–	–	Farbmonitor, Drucker, Software
Sharp	PC-1401	340	120	160	500	CE 126, CE 152, diverse Software
	PC-1500	520	1000 bis 400	450	1400 bis 400	CE 150, CE 155, Software, Literatur
Sinclair	ZX Spectrum	250	750 bis 300	–	–	2 Laufwerke, Drucker, Cartridges, Software
	QL	570	980	1250 bis 1680	1890	128 KB, Monitor, 2 Microdrives
Texas Instruments	TI-99/4A	450	190	198 bis 390	285 bis 3500	Kassettenrecorder, Monitor, V.24-Schnittstelle, diverse Software, Literatur
Triumph Adler	Alphatronic PC	1000	950 bis 2800	1200 bis 2700	–	–

Das ist Ihr Computer heute wert: *Unsere Vergleichstabelle zeigt die Gebrauchtpreislage der gängigsten Systeme aus den letzten drei Jahren.*

Die digitalen Reisen des Hacker S.

Unser fünfter Teil beschäftigt sich mit dem Hacken in internationalen Rechnern

Das Energieministerium, die staatliche Forstbehörde, die Nationalbibliothek sowie zahlreiche Firmen aus Neuseeland bekamen Ende April Besuch aus Deutschland: Mad Max, ein 17jähriger Schüler aus Frankfurt, hatte auf seinem C64 sechzehn Stunden lang einen NUA-Scanner laufen lassen.

NUA-Scanner

Scanner sind simple Programme, die in Netzknoten (Datex, Tymnet, usw.) Anwahlnummern von Computern durchprobieren. Ihre logische Struktur demonstriert dieses Programm in Pseudo-Pascal:

```
Loop:     Nummer = Nummer
          + 1
          OUTPUT Nummer TO
          RS232
          IF INPUT RS232 <>
          "Not available"
          THEN Schreib
          GOTO Loop
Schreib:  PRINT Nummer,
          INPUT RS232
          OUTPUT "(CTRL P)
          CLR" TO
          RS232
          Wait 500
          GOTO Loop
```

Nummern werden durchprobiert, Rechner, und Knotenmeldungen abgefangen, bei interessanten Meldungen wird mitprotokolliert und dann die Verbindung wieder gelöst (Control-P CLR ist der entsprechende Befehl im Datex-Netz). Gehackt wird dann am nächsten Morgen anhand der gefundenen Nummern von Hand.

Die neuseeländischen Nummern wanderten natürlich schnell in Hackerkreisen. Und weil hundert Hacker mehr Passwörter ausprobieren können als einer allein, waren die ersten Codes auch bald geknackt. Forstverwaltung und Energieministerium dementierten bei Bekanntwerden natürlich schleunigst, daß Mad Max an irgendwelche sicherheitsrelevanten Daten herangekommen sei. Doch alle Betroffenen änderten sicherheitshalber schleunigst ihre Passwörter. (Wer selber mal nach Neuseeland gehen will, dem sei die NUA 0530197000016 empfohlen. User-ID ist „MATE", Passwort „VISITOR".)

„Free flow of information"

„Intergalaktische Vereinigungen ohne feste Strukturen" sind die bundesdeutschen Hackerclubs nach eigenem Selbstverständnis. Lautstark mahnen sie einen umfassenden Schutz der persönlichen Daten an, und betrachten sich selbst als „Verbraucherschützer auf den Netzen". Statt einer immer weiter um sich greifenden Monopolisierung von Netzwerken und Informationsbanken fordern sie den free flow of information, den freien und ungehinderten Informationsaustausch: Freie Kommunikation zwischen allen und jedem, Zugang zu allen gesellschaftlich wichtigen Daten und Demokratisierung der Netze. Für sich selber nehmen die Computerfreaks diese Forderungen bereits vorweg: Als Hacker S. vor einigen Wochen amerikanische Mailboxen besuchte, kam er zu einem hochinteressanten Chat.

Chatten

In manchen Mailboxen und in großen Rechnersystemen können mehr als nur ein Benutzer gleichzeitig eingeloggt sein. Das Gespräch untereinander in einem Rechner, das Chatten, ist für Hacker die beliebteste Art, um neue Leute kennenzulernen und Informationen auszutauschen. Wer's ausprobieren will: NUA 03106000102. Mit „finger" läßt man sich die User ausgeben, die gerade online sind, mit „whois" kann man abfragen, wer sie sind.

Wie sich bei dem mehrstündigen Chatten herausstellte, war „Saratoga" – so nannte sich der Hacker – Wachoffizier auf dem US-amerikanischen Flugzeugträger „Saratoga", der vor Lybien kreuzte. Da die Bombardierung schon ein paar Tage vorbei war, langweilte er sich schrecklich und benutzte seinen mitgebrachten Apple II und den militärischen Nachrichtensatelliten, mit dem die US-Navy die Verbindungen zu ihren Schiffen aufrechthält, dazu, sich in der Weltgeschichte herumzutreiben.

Je mehr Computer es gibt, um so mehr Hacker bringt das Land hervor. Zwei Drittel aller internationalen Datenreisenden sind Amerikaner. Je zehn Prozent entfallen auf Engländer und Deutsche, der Rest der Welt ist nur sporadisch vertreten. So war es reiner Zufall, daß unser Hacker S. Anfang des Jahres „UJC", einen Japaner, in einer amerikanischen VAX traf, der sich enorm freute, einen Deutschen zu treffen, denn so erzählte er, er besitze ein deutsches Auto. Es sei – und dabei blieb er hartnäckig – ein Melcedes.

Nicht jeder Rechnerbetreiber schätzt allerdings Freiheit und Demokratie auf den Netzen. Vor einiger Zeit teilte die Cornell-Universität aus New York den Rechenzentren der westlichen Welt mit, daß

ihre Verbindungen gekappt würden, da europäische Hacker das System unsicher machen würden. Hintergrund der Aktion: Fermilab, eine Großforschungseinrichtung südwestlich von Chikago, wurde von den Kids zum Chatten verwendet. Da an Cornell auch Arpanet, das amerikanische Militärnetz, hängt, haben gut unterrichteten Kreisen zufolge die kommunistenängstlichen Amerikaner im Zuge ihres SDI-Programms die Hackeraktivitäten als Vorwand genommen, um die Europäer von amerikanischen Informationen abzuhängen. Wie heißt es so schön in der Datenschleuder Nr. 7 vom Chaos-Computer-Club: ,,Je sicherer ein System gegen unbefugten Zugriff geschützt wird, desto unbefugter müssen diejenigen sein, die die Schwachstellen aufdecken. Am Ende kommt dabei ein System heraus, auf das überhaupt nur noch die Unbefugten Zugriff haben." Wen es danach drängt, sich mit amerikanischen Sicherheitsproblemen rumzuschlagen, dem empfiehlt Hacker S. die NUA 0311050500061. Das ist das Los Alamos Integrated Computing Network, von dem aus man sich mit ,,DNA" zur Defense Nuclear Ageny weiterverbinden lassen kann. Eine interessante Nacht kann bevorstehen.

Europa

OVID, das Datenverarbeitungssystem des Europäischen Parlaments hängt übrigens auch am Datex-Netz: Nach der NUA 027042920101 gibt man eine ,,5" und ein ⟨RETURN⟩ ein. Danach kann man sich Sitzungskalender und -protokolle abrufen sowie nach Stichworten oder Abgeordneten fahnden. Selbstverständlich ist das Gerät neunsprachig, nur die Systemmeldungen kommen immer in französisch (Interessant zum Beispiel die Sitzungsprotokolle zum Thema ,,Gewalt gegen Frauen").

Wer sich allerdings im Dickicht der Netze und Netzknoten durchfinden will, der braucht schon ein bißchen mehr als eine stattliche Sammlung von NUA's. Oft ist es gar nicht so gut, sich in einen Rechner direkt einzuloggen, ein Netzknoten wie Tymnet taugt nicht nur zum Spurenverwischen.

Alle Knoten dieser Erde

Hinter der amerikanischen NUA 0311020200141 verbirgt sich ein solcher Knoten: Telenet betreibt hier unter anderem einen Phone-Service und ein ,,International Information System". Mit der User-ID ,,PHONES" und Passwort ,,PHONES" sind sämtliche lokalen Telenet-Telefonnummern abrufbar. Gibt man als User-ID ,,INTL" und als Organisation ,,ASSOCIATES" ein (das Passwort ist ,,INTL"), kann er Informationen über die Datennetzdienste sämtlicher Länder der Erde, die über so etwas verfügen, abrufen. Auch Sachen wie ,,X.121 Numbering Format" oder ,,International Trouble Reporting Procedures" sind verfügbar.

```
-O.S.T............01
Communication etablie
 VOTRE NUMERO D IDENTIFICATION SVP
5
 5

 VEUILLEZ PATIENTER SVP

 EXPERIMENTAL  INSIS  VIDEOTEX  SYSTEM

     BITTE WAEHLEN SIE EINE SPRACHE
     PLEASE SELECT A LANGUAGE
     CHOISISSEZ UNE LANGUE S.V.P.
     PER FAVORE SCEGLIERE UNA LINGUA
              1 DEUTSCH
              2 ENGLISH
              3 FRANCAIS
              4 ITALIANO

 Teletel:Nr -> <ENVOI>  Btx:Nr
 Prestel:Nr             TTY:Nr -> <CR>
              O V I D E
                                  100a1

 1
 VEUILLEZ PATIENTER SVP

             AUSKUENFTE ZUR VERBES-
             SERUNG DES DIENSTES

 WER SIND SIE ?
 1-Abgeordneter        2-Assistent e.Abg.
 3-Fraktionsbeamter    4-Beamter d. EP
 5-Kommissionsbeamter 9-Sonstige
 VON WO RUFEN SIE AN ?
 B-Belgien   DK-Daenemark IRL-Irland
 D-Deutschld GR-Griechenld  L-Luxemburg
 F-Frankr.   NL-Niederlde   E-Spanien
 I-Italien   UK-Grossbrit.  P-Portugal
 IHRE MUTTERSPRACHE ?
 DA,DE,EL(griechisch),EN,ES,FR,IT,NE,PO

 Ihre Antwort getrennt durch Leerzeichen
 dann: Teletel:ENVOI      Btx:#
       Prestel:#          TTY:CR
 OVIDE DEUTSCH
                                332331a
1 F PO
                               1 F PO

VEUILLEZ PATIENTER SVP

             HAUPTMENU

 1 SITZUNGSKALENDER-TAGESORDNUNGEN
 2 PUBLIKATIONEN-DOKUMENTATION
   !  EUROSTAT *20026
 3 MITGLIEDERLISTE  STELLENPLAN EP
 4 PRAKTISCHE HINWEISE

 5 MITTEILUNGSDIENST (FUER TTY: 6)
 7 VERZEICHNIS DER TEILNEHMER
 8 IHRE KRITIK UND IHRE VORSCHLAEGE

IHRE WAHL      Teletel:Nr ENV  Btx:Nr
               Prestel:Nr      TTY:Nr CR
BEDIENGSANLTG:Teletel:GUIDE    Btx:*1#
               Prestel:*1#     TTY:/G/CR

                                  200a1
```

Hacking International:
In verschiedenen Sprachen kann man sich 4 Informationen des Europäischen Parlaments abrufen

Autodial ist auch eine feine Sache, die die internationale Verständigung belebt und die Telefonkosten dabei niedrig hält. Und wer schon einmal in einer amerikanischen Phone-Mailbox drin war und mit dem Sysop gechattet hat, der weiß, was für ein erhebendes Gefühl es ist, wenn der arme Mensch an anderen Ende der Leitung gar nicht glauben will, daß man aus ,,Germany" anruft.

Unter einem Autodial stellt man sich am besten das Gegenstück zu einem Datex-Pad vor. Der Pad ist ein Rechner, der per Telefon angerufen wird und der dann in ein Datennetz schaltet. Ein Autodialer macht es gerade umgekehrt: Man kommt per Datennetz und wird ins Telefonnetz weitergeschaltet. Wobei dann natürlich nur Computer angerufen werden können. Stimmen überträgt kein Datennetz. Je nachdem, was für Accounts und Passwörter gerade im Umlauf sind, kann es recht kostensparend sein, sich unter Umständen ein paar mal hin- und herzuschalten. Und welcher gestreßte Postbeamte kann schon eine Spur weiterverfolgen, die von einer deutschen Amtsleitung ins Datex-Netz geht, von dort zu Tymnet nach England, von dort per Satellit in einen (oder mehrere) amerikanischen Rechner und schließlich in einem der vielen amerikanischen Telefonnetze endet. Sofern der Hacker es nicht vorzieht, mit dem Autodial wieder eine deutsche Nummer anzuwählen, um endlich mal zum Ortstarif eine Telefonmailbox anzurufen, die wesentlich außerhalb des Achtminutentakts liegt.

Hacker S. findet nicht nur viele Freunde auf seinen Streifzügen durch den Netzdschungel. Auch kilobyteweise Informationen liegen dort für ihn bereit, manches ist sogar von den Verfassern zum freien Gebrauch bestimmt. Die interessantesten Datenbanken und wie man sie benützt, werden im sechsten Teil der digitalen Reisen des Hackers S. beschrieben.

Joachim Graf

Aktiv Computern

Listing des Monats Reversi

Das Programm variiert die Original-Spielregeln, indem es pro Zug nur eine Richtung zuläßt, die jeweils ausgewählt werden muß (CPC 464).

Das Programm gestattet Ihnen, in dem bekannten gleichnamigen Spiel gegen ihren Schneider-Computer anzutreten. Gespielt wird auf einem 8 × 8 Felder großen Spielfeld. Ein Spieler erhält rote, der andere grüne Spielsteine. Die Spieler setzen nun abwechselnd jeweils einen ihrer Steine so auf das Spielfeld, daß dieser eine zusammenhängende gerade Reihe des Gegners zusammen mit einem eigenen, bereits gesetzten Stein einschließt. Dies kann in alle Richtungen, senkrecht, waagrecht oder diagonal geschehen. Die eingeschlossenen Steine des Gegners wechseln dann ihre Farbe, nehmen also die eigene Farbe an. Das Spiel ist beendet, wenn auf diese Weise kein Zug mehr möglich ist. Gewonnen hat der Spieler, der die meisten Steine seiner Farbe auf dem Spielfeld liegen hat.

Nach dem Starten des BASIC-Programmes mit RUN wird ein Maschinenprogramm aus DATA-Zeilen in den Speicher gelesen. Alle Eingaben im Spiel geschehen ausschießlich über den Joystick oder das Cursortastenfeld (wobei die COPY-Taste als Feuerknopf dient). Zunächst müssen Sie wählen, ob Sie gegen den Computer spielen wollen oder bei einem DEMO-Spiel zuschauen möchten. Bringen Sie dazu den Cursor (roter Pfeil) mit dem Joystick in die entsprechende Stellung und drücken den Feuerknopf.

Spiel gegen den Computer

Wählen Sie zunächst aus, mit welcher Farbe Sie spielen wollen, ob Sie den ersten Zug machen möchten und in welcher Spielstufe (0–3) der Computer spielen soll. Die Spielstufe gibt an, wieviele Züge der Computer vorausdenkt. Die angegebenen Bedenkzeiten sind Durchschnittszeiten pro Zug und gelten nur für den Computer; Sie selbst können sich beliebig viel Zeit für einen Zug lassen. Sind alle Eingaben gemacht, wird das Spielfeld angezeigt, und das Spiel kann beginnen. Um einen Stein zu setzen, fahren Sie mit dem Cursor (schwarzes oder weißes Kreuz) auf das entsprechende Feld und drücken den Feuerknopf. Ist ein Einschließen der gegnerischen Steine nach mehreren Seiten möglich, müssen Sie zusätzlich noch die Richtung angeben. Fahren Sie dazu den Cursor auf das entsprechende der acht Nachbarfelder und drücken den Feuerknopf.

Während des Spiels stehen Ihnen fünf Menüpunkte zur Verfügung. Um ins Menü zu gelangen, fahren Sie den Cursor einfach bei einem der fünf Pfeile links aus dem Spielfeld heraus. Der entsprechende Menüpunkt erscheint dann in inverser Schrift. Auch im Menü können Sie den Cursor auf- und abwärts bewegen. Bei dem gewünschten Menüpunkt drücken Sie dann den Feuerknopf. Um wieder ins Spielfeld zu gelangen, be-

```
1 REM %%%%%%%%%%%%%%%%%%%%%%
2 REM %                    %
3 REM %  *** REVERSI ***   %
4 REM %                    %
5 REM %  September  1985   %
6 REM %                    %
7 REM %%%%%%%%%%%%%%%%%%%%%%
8 :
10 ON ERROR GOTO 7000
20 SYMBOL AFTER 256:MEMORY &8FFF:POKE &9
000,0
30 CLEAR:RANDOMIZE TIME
40 ON ERROR GOTO 7000:ON BREAK GOSUB 30
50 DEFINT a-z:DEFREAL l,p:DIM f(7,7),zu(
128),s(8),m$(5),st(2)
60 MODE 1:INK 0,0:INK 1,3:INK 2,9:INK 3,
13:PAPER 0:BORDER 0:CLS
61  REM Gruenmonitor: INK 1,9:INK 2,26:I
NK 3,18
70 FOR j=0 TO 79:KEY DEF j,0,&FF,&FF,&FF
:NEXT
80 KEY DEF 0,1,11,11,11:KEY DEF 1,1,9,9,
9:KEY DEF 2,1,10,10,10:KEY DEF 8,1,8,8,8
:KEY DEF 9,0,42,42,42
90 KEY DEF 72,1,11,11,11:KEY DEF 75,1,9,
9,9:KEY DEF 73,1,10,10,10:KEY DEF 74,1,8
,8,8:KEY DEF 76,0,42,42,42:KEY DEF 77,0,
42,42,42
100 KEY DEF 48,1,11,11,11:KEY DEF 51,1,9
,9,9:KEY DEF 49,1,10,10,10:KEY DEF 50,1,
8,8,8:KEY DEF 52,0,42,42,42:KEY DEF 53,0
,42,42,42
110 KEY DEF 66,0,252,252,252:SPEED KEY 1
6,8:SYMBOL AFTER 240
120 RESTORE 10100:FOR i=0 TO 10
130 FOR j=0 TO 8:READ a$:s(j)=VAL("&"+a$
):NEXT
140 SYMBOL s(0),s(1),s(2),s(3),s(4),s(5)
,s(6),s(7),s(8)
150 NEXT
160 PEN 1:LOCATE 13,2:PRINT"R E V E R S
I":LOCATE 24,4:PEN 2:PRINT CHR$(164);" 1
985":PEN 3
170 IF PEEK(&9000) THEN 220
180 PRINT CHR$(10);CHR$(10);"Bitte warte
n !"
190 PRINT"Maschinenprogramm wird eingele
sen."
200 GOSUB 8000:REM $9000-$939F
210 PRINT"O.k."
220 PEN 2:PRINT STRING$(3,10);"  SPIELMO
DUS"
230 PEN 1:PRINT CHR$(10);CHR$(10);"  1.
";:PEN 3:PRINT"Du spielst gegen mich";:v
=VPOS(#0)
240 PRINT CHR$(10);"  2. DEMO: Ich spiel
e mit mir selber"
250 sm=1:GOSUB 6060:WHILE NOT te
260 LOCATE 1,v+2*(sm-1):PEN 1:PRINT"> ";
HEX$(sm);"."
270 GOSUB 6000:IF NOT(ta XOR tu) THEN 30
0
280 LOCATE 1,v+2*(sm-1):PEN 3:PRINT"  ";
HEX$(sm);"."
290 sm=(1 AND ta)+(2 AND tu)
300 WEND:IF sm=2 THEN 500
310 CLS:PEN 3:PRINT"Willst Du mit den ";
:PEN 1:PRINT"roten";:PEN 3:PRINT" oder "
;:PEN 2:PRINT"gr";CHR$(255);"nen":PEN 3:
PRINT"Steinen spielen ?"
320 PEN 1:PRINT CHR$(10);">";:PEN 3:PRIN
T" rot":PRINT"  gr";CHR$(255);"n"
330 f=1:GOSUB 6060:WHILE NOT te
340 LOCATE 1,3+f:PEN 1:PRINT">"
350 GOSUB 6000:IF NOT(ta XOR tu) THEN 38
0
360 LOCATE 1,3+f:PRINT" "
370 f=(1 AND ta)+(2 AND tu)
```

wegen Sie einfach den Cursor nach rechts.

Passen
Ist ein Spieler nicht in der Lage, einen erlaubten Zug zu machen, so muß er „passen", und der andere Spieler kommt wieder an die Reihe. Passen ist jedoch nur erlaubt, wenn tatsächlich kein Zug mehr möglich ist.

Tip
Der Computer sagt Ihnen, welchen Zug er an Ihrer Stelle tun würde. Zeigt er mehrere Züge an, dann betrachtet er diese als gleichwertig. Es ist Ihnen freigestellt, diesen oder einen anderen Zug zu machen.

Seitenwechsel
Der Computer spielt mit Ihren und Sie mit seinen Steinen weiter.

Zug zurücknehmen
Der letzte Zug im Spiel wird zurückgenommen. Dieser Menüpunkt kann so oft aufgerufen werden, bis sich das Spiel wieder in der Ausgangsstellung befindet. Hinweis: Bei „Seitenwechsel" und „Zug zurücknehmen" ist es möglich, daß nicht mehr Sie, sondern der Computer am Zug ist. Dieser Fall wird dann angezeigt. Sie können dann entweder noch weitere Züge zurückspielen, oder, wenn der Computer den Zug ausführen soll, mit dem Cursor auf ein beliebiges Feld auf dem Spielfeld fahren und den Feuerknopf drücken.

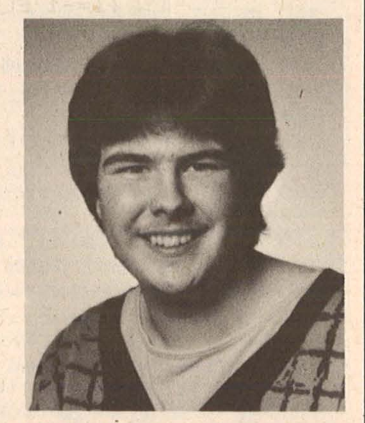
Spielstufe ändern
Auch während des Spiels kann die Spielstufe geändert werden. Sie kann zum Beispiel für einen Tip kurzfristig erhöht werden. Fahren Sie einfach den Cursor auf die gewünschte Zahl und drücken Sie den Feuerknopf. Während des Spiels werden jeweils die 19 letzten Doppelzüge rechts angezeigt. Das Spiel kann durch zweimaliges Drücken der ESC-Taste auch abgebrochen werden, jedoch nicht, solange der Text „Ich überlege." angezeigt wird. Beim Abbruch erscheint wieder das Titelbild, und Sie können erneut ein Spiel beginnen. Das Spiel ist regulär beendet, wenn das Spielfeld voll ist oder kein Zug mehr möglich ist. In diesem Fall wird das Endergebnis angezeigt. Durch Drücken des Feuerknopfes können Sie ein neues Spiel starten.

DEMO-Spiel
Wählen Sie zunächst aus, in welchen Spielstufen die Gegner jeweils spielen sollen. Anschließend brauchen Sie nichts mehr einzugeben und können das Spiel verfolgen. Es werden jeweils das Spielfeld und die 19 letzten Doppelzüge angezeigt. Ein Eingreifen in das Spiel ist nicht möglich. Sie können es lediglich, wie oben beschrieben, abbrechen.

Michael Knaup

Der Autor stellt sich vor

Obwohl mein Lebenslauf nach 17 Jahren noch nicht allzu lang ist, möchte ich mich doch kurz vorstellen: 1979 habe ich mit Erfolg die Grundschule beendet, inzwischen besuche ich die 11. Klasse Gymnasium. Meine ersten Erfahrungen mit Computern machte ich mit fünf Jahren auf einem Wang 600 – einem der ersten Kleincomputer überhaupt. BASIC lernte ich erst später mit Hilfe des programmierbaren Taschenrechners Sharp PC-1211. Seit eineinhalb Jahren beschäftige ich mich nun intensiv mit dem CPC 464. Meine Hobbys sind neben Computern noch Orgelspielen, Fotografieren und Arbeiten mit Holz.



Michael Knaup, Autor von Reversi

```
380 WEND
390 LOCATE 1,8:PEN 3:PRINT"Willst Du den
 ersten Zug machen ?"
400 PEN 1:PRINT CHR$(10);">";:PEN 3:PRIN
T" ja":PRINT"  nein"
410 ss=1:GOSUB 6060:WHILE NOT te
420 LOCATE 1,9+ss:PEN 1:PRINT">"
430 GOSUB 6000:IF NOT(ta XOR tu) THEN 46
0
440 LOCATE 1,9+ss:PRINT" "
450 ss=(1 AND ta)+(2 AND tu)
460 WEND
470 sc=ss XOR 3:s(ss)=f:s(sc)=f XOR 3
500 CLS:PEN 2:ZONE 15:PRINT"SPIELSTUFE",
"Bedenkzeit pro Zug ca."
510 PEN 1:PRINT CHR$(10);">  0",:PEN 3:P
RINT" ";CHR$(169);" sec."
520 PRINT"   1"," 3 sec."
530 PRINT"   2","30 sec."
540 PRINT"   3","10 min."
550 IF sm=1 THEN GOSUB 900:GOTO 600
560 s(1)=1:s(2)=2:FOR j=1 TO 2
570 LOCATE 1,8:PRINT"Bitte Spielstufe f"
;CHR$(255);"r Spieler";j;:IF j=1 THEN PE
N 1:PRINT"(rot)" ELSE PEN 2:PRINT"(gr";C
HR$(255);"n)"
580 PEN 3:PRINT"eingeben !":GOSUB 900:st
(j)=st:LOCATE 1,3+st:PEN 3:PRINT"  ";st
590 NEXT
600 CLS:INK 3,0
610 ORIGIN 152,56,152,466,56,374:CLG 3
620 PAPER 3:PEN 0:FOR j=1 TO 8
630 LOCATE 11+2*j,3:PRINT CHR$(64+j)
640 LOCATE 11+2*j,21:PRINT CHR$(64+j)
650 LOCATE 11,3+2*j:PRINT CHR$(48+j)
660 LOCATE 29,3+2*j:PRINT CHR$(48+j)
670 NEXT
680 FOR l=-5.5 TO 5.5
690 sl=CINT(SQR(36-l^2))*2
700 FOR i=326 TO 102 STEP-32
710 FOR j=200 TO 424 STEP 32
720 ORIGIN j,i
730 MOVE -sl,-INT(l)*2:DRAWR 2*sl-2,0,0
740 NEXT j,i,l:INK 3,13
741  REM Gruenmonitor: INK 3,18
750 x=3:y=3:f=s(1):f(x,y)=1:GOSUB 6100
760 x=4:y=3:f=s(1):f(x,y)=1:GOSUB 6100
770 x=3:y=4:f=s(2):f(x,y)=2:GOSUB 6100
780 x=4:y=4:f=s(2):f(x,y)=2:GOSUB 6100
790 q=2:s(0)=0
800 ON sm GOTO 1000,1500
810 :
900 st=0:GOSUB 6060:WHILE NOT te
910 LOCATE 1,3+st:PEN 1:PRINT"> ";st
920 GOSUB 6000:IF NOT(ta XOR tu) THEN 95
0
930 LOCATE 1,3+st:PEN 3:PRINT"  ";st
940 st=(st+ta-tu) AND 3
950 WEND:RETURN
1000 :
1001 REM %%%%%%%%%%%%%%%%%%%%%%%%%
1002 REM % Spiel gegen Computer %
1003 REM %%%%%%%%%%%%%%%%%%%%%%%%%
1004 :
1010 PAPER 0:PEN 2:LOCATE 14,1:PRINT"Spi
elstufe ";:PEN 3:PRINT st
1020 PEN 2:LOCATE 1,3:PRINT"MENUE"
1030 mlf$=CHR$(13)+CHR$(10):m$(1)="Passe
n":m$(2)="Tip":m$(3)="Seiten-"+mlf$+"wec
hsel":m$(4)="Zug"+mlf$+"zur"+CHR$(255)+"
ck-"+mlf$+"nehmen":m$(5)="Spiel-"+mlf$+"
stufe"+mlf$+CHR$(253)+"ndern":FOR j=1 TO
 5:m$(j)=m$(j)+mlf$+CHR$(10):NEXT:m$(3)=
m$(3)+CHR$(10)
1040 LOCATE 1,5:FOR j=1 TO 5
1050 PEN 2:PRINT TAB(9);CHR$(242);CHR$(1
3);:PEN 3:PRINT m$(j);
1060 NEXT
```

```
1070 WINDOW#1,1,40,24,25:PAPER#1,0:PEN #
1,3:CLS#1
1080 WINDOW#2,31,40,3,23:PAPER#2,0:CLS#2
1100 WHILE s<60 AND fp<2
1110 q=q XOR 3
1120 ON (q=ss)+2 GOSUB 3000,4000
1130 CLS#1:GOSUB 2000
1140 WEND
1150 GOTO 5000
1500 :
1501 REM %%%%%%%%%%%%%%
1502 REM % Demo-Spiel %
1503 REM %%%%%%%%%%%%%%
1504 :
1510 PAPER 0:LOCATE 1,3:PEN 1:PRINT"Spie
ler 1":LOCATE 1,5:PEN 3:PRINT"Stufe";st(
1)
1520 LOCATE 1,9:PEN 2:PRINT"Spieler 2":L
OCATE 1,11:PEN 3:PRINT"Stufe";st(2)
1530 LOCATE 15,1:PEN 2:PRINT"DEMO - SPIE
L"
1540 WINDOW#1,1,40,24,25:PAPER#1,0:PEN #
1,3:CLS#1
1550 WINDOW#2,31,40,3,23:PAPER#2,0:CLS#2
1600 WHILE s<60 AND fp<2
1610 q=q XOR 3
1620 sc=q:st=st(q):GOSUB 4000
1630 CLS#1:GOSUB 2000
1640 WEND
1650 GOTO 5000
2000 :
2001 REM %%%%%%%%%%%%%%%%%%%
2002 REM % Zug ausfuehren %
2003 REM %%%%%%%%%%%%%%%%%%%
2004 :
2010 zu=zu+1:zu(zu)=z AND 63
2020 IF zu MOD 2 THEN PEN#2,3:PRINT#2,US
ING "##";CINT(zu/2);
2030 IF z=512 THEN fp=fp+1:a$=" - ":zu(
zu)=-1:GOTO 2100
2040 x=(z AND 56)/8:y=z AND 7:z=(z AND 4
48)/64
2050 fp=0:s=s+1:a$=" "+CHR$(65+x)+CHR$(4
9+y)+CHR$(240+z)
2060 FOR j=1 TO 5:f=s(q)*(j MOD 2):GOSUB
 6100:NEXT:f(x,y)=q
2070 RESTORE:FOR j=1 TO z+1:READ rh,rv:N
EXT
2080 x=x+rh:y=y+rv:IF f(x,y)=(q XOR 3) T
HEN f(x,y)=q:f=s(q):GOSUB 6100:GOTO 2080
2090 zu(zu)=zu(zu)+512*x+64*y
2100 PEN#2,s(q):PRINT#2,a$;:IF q=1 THEN
PRINT#2,CHR$(10);CHR$(10);CHR$(11);CHR$(
11);
2110 RETURN
3000 :
3001 REM %%%%%%%%%%%%%%%%
3002 REM % Spieler Zug %
3003 REM %%%%%%%%%%%%%%%%
3004 :
3010 DEF FN menue(x)=x+1-(CINT(x/4) AND
(x>2)):x=3:y=3:CLS#1:PRINT#1,CHR$(7);CHR
$(10);"Bitte Zug eingeben !"
3020 GOSUB 6060:WHILE NOT te
3030 IF x=-1 THEN PAPER 3:PEN 1:LOCATE 1
,5+2*y:m=FN menue(y):PRINT m$(m)
3040 IF x>-1 THEN PAPER s(f(x,y)):PEN 3-
3*SGN(f(x,y)):LOCATE 13+2*x,5+2*y:PRINT
CHR$(159)
3050 GOSUB 6000
3060 IF NOT(ta OR tu OR tl OR tr OR te)
THEN 3130
3070 IF x=-1 THEN PAPER 0:PEN 3:LOCATE 1
,5+2*y:m=FN menue(y):PRINT m$(m)
3080 IF x>-1 THEN PAPER s(f(x,y)):LOCATE
 13+2*x,5+2*y:PRINT" "
3090 x=x+tl-tr
3100 IF x<-1 THEN x=-1 ELSE IF x>7 THEN
x=0
3110 y=(y+ta-tu) AND 7
3120 IF x=-1 AND (y=3 OR y=5 OR y=7) THE
N IF ta XOR tu THEN 3110 ELSE x=0
3130 WEND
3140 IF x=-1 THEN ON FN menue(y) GOTO 35
00,3600,3700,3800,3900
3150 IF q=sc THEN 4000
3160 IF f(x,y)<>0 THEN CLS#1:PRINT#1,CHR
$(7);CHR$(10);"Feld schon besetzt !":GOT
O 3020
3170 RESTORE:z=-1:fm=0:FOR j=0 TO 7
3180 READ rh,rv:a=x:b=y:fz=0
3190 a=a+rh:b=b+rv
3200 IF a<0 OR a>7 OR b<0 OR b>7 THEN 32
40
3210 IF f(a,b)=0 THEN 3240
3220 IF f(a,b)=sc THEN fz=-1:GOTO 3190
3230 IF fz THEN IF z>-1 THEN fm=-1:j=7 E
LSE z=j
3240 NEXT:IF z=-1 THEN CLS#1:PRINT#1,CHR
$(7);CHR$(10);"Unerlaubter Zug !":GOTO 3
020
3250 IF NOT fm THEN 3450
3260 CLS#1:PRINT#1,CHR$(7);CHR$(10);"Bit
te Richtung angeben !"
3270 LOCATE 13+2*x,5+2*y:PAPER 0:PEN s(s
s):PRINT CHR$(231)
3280 fl=0:GOSUB 6060:FOR i=1 TO 0 STEP-1
3290 RESTORE:FOR j=0 TO 7
3300 READ rh,rv:a=x+rh:b=y+rv
3310 IF a>=0 AND a<8 AND b>=0 AND b<8 TH
EN LOCATE 13+2*a,5+2*b:PAPER s(f(x+rh,y+
rv)):PEN i*3+(1-i)*s(f(x+rh,y+rv)):PRINT
 CHR$(240+j)
3320 NEXT:FOR j=1 TO 10:IF INKEY$<>"" OR
 fl THEN j=10:fl=-1
3330 NEXT j,i:IF NOT fl THEN 3280
3340 a=x:b=y:GOSUB 6060:WHILE NOT te
3350 LOCATE 13+2*a,5+2*b:PAPER s(f(a,b))
:PEN 3-3*SGN(f(a,b)):PRINT CHR$(159)
3360 GOSUB 6000:IF NOT(ta OR tu OR tl OR
 tr OR te) THEN 3390
3370 LOCATE 13+2*a,5+2*b:IF a=x AND b=y
THEN PAPER 0:PEN s(ss):PRINT CHR$(231) E
LSE PAPER s(f(a,b)):PRINT" "
3380 a=a+tl-tr:a=a+(a>7)-(a<0):a=a+(a>x+
1)-(a<x-1):b=b+ta-tu:b=b+(b>7)-(b<0):b=b
+(b>y+1)-(b<y-1)
3390 WEND:IF a=x AND b=y THEN 3340
3400 RESTORE:FOR j=0 TO 7
3410 READ rh,rv:IF a-x=rh AND b-y=rv THE
N z=j:j=7
3420 NEXT:a=x:b=y:fz=0:fl=0
3430 a=a+rh:b=b+rv:IF a<0 OR a>7 OR b<0
OR b>7 THEN fl=-1 ELSE IF f(a,b)=0 THEN
fl=-1 ELSE IF f(a,b)=sc THEN fz=-1:GOTO
3430 ELSE IF NOT fz THEN fl=-1
3440 IF fl THEN CLS#1:PRINT#1,CHR$(7);"U
nerlaubter Zug !":PRINT#1,"Bitte Richtun
g angeben !":GOTO 3340
3450 z=z*64+x*8+y
3460 RETURN
3500 :
3501 REM  ***
3502 REM  *** Passen
3503 REM  ***
3504 :
3510 sc=sc XOR 3:st(1)=st:st=0:GOSUB 402
0:st=st(1):sc=sc XOR 3
3520 IF z=512 THEN CLS#1:PRINT#1,CHR$(7)
;CHR$(10);"O.k.":RETURN
3530 CLS#1:PRINT#1,CHR$(7);CHR$(10);"Nur
 passen, wenn Ziehen unm";CHR$(254);"gli
ch !"
3540 x=-1:y=0:GOTO 3020
3600 :
3601 REM  ***
```

```
3602 REM  *** Tip
3603 REM  ***
3604 :
3610 sc=sc XOR 3:GOSUB 4000:sc=sc XOR 3
3620 CLS#1:PRINT#1,CHR$(7);CHR$(10);"Mei
n Tip:  ";
3630 PEN#1,s(ss):z=PEEK(&A009):IF z=0 TH
EN PRINT#1,"Passen":GOTO 3680
3640 FOR j=0 TO (z-1)*2 STEP 2
3650 z=PEEK(&A200+j)+256*PEEK(&A201+j)
3660 PRINT#1,CHR$(65+(z AND 56)/8);CHR$(
49+(z AND 7));CHR$(240+(z AND 448)/64);"
 ";
3670 NEXT
3680 PEN#1,3:x=-1:y=1:GOTO 3020
3700 :
3701 REM  ***
3702 REM  *** Seitenwechsel
3703 REM  ***
3704 :
3710 ss=sc:sc=sc XOR 3
3720 CLS#1:PRINT#1,CHR$(7);CHR$(10):"O.k
. Du spielst jetzt mit ";
3730 ON s(ss) GOSUB 3780,3790
3740 PRINT#1," weiter."
3750 IF q=sc THEN PEN#1,1:PRINT#1,"Achtu
ng";:PEN#1,3:PRINT#1,", ich bin am Zug !
"
3760 GOTO 3020
3780 PEN#1,1:PRINT#1,"rot";:PEN#1,3:RETU
RN
3790 PEN#1,2:PRINT#1,"gr";CHR$(255);"n";
:PEN#1,3:RETURN
3800 :
3801 REM  ***
3802 REM  *** Zug zuruecknehmen
3803 REM  ***
3804 :
3810 IF zu=0 THEN CLS#1:PRINT#1,CHR$(7);
CHR$(10);"Spiel bereits in Ausgangsstell
ung !":GOTO 3020
3820 IF zu(zu)=-1 THEN 3840 ELSE GOSUB 3
890:f(x,y)=0:f=0:GOSUB 6100
3830 WHILE x<>x2-rh OR y<>y2-rv:x=x+rh:y
=y+rv:f(x,y)=q:f=s(q):GOSUB 6100:WEND
3840 x=7-6*(zu MOD 2):y=CINT(zu/2):IF y>
19 THEN y=19
3850 LOCATE#2,x,y:PRINT#2,CHR$(18):LOCAT
E#2,x,y
3860 IF (zu MOD 2)=1 AND zu>38 THEN zu=z
u-38:PEN#2,3:PRINT#2,CHR$(30);CHR$(11);:
PRINT#2,USING"##";CINT(zu/2);:GOSUB 3890
:PEN#2,s(1):GOSUB 3880:zu=zu+1:GOSUB 389
0:PEN#2,s(2):GOSUB 3880:zu=zu+37:LOCATE#
2,1,20
3870 zu=zu-1:s=s-1:q=q XOR 3:x=-1:y=4:CL
S#1:PRINT#1,CHR$(7);CHR$(10);"O.k.":GOTO
 3750
3880 RESTORE:z=-1:rh1=0:rh2=0:WHILE rh<>
rh1 OR rv<>rv1:READ rh1,rv1:z=z+1:WEND:P
RINT#2," ";CHR$(65+x);CHR$(49+y);CHR$(24
0+z);:RETURN
3890 z=zu(zu):x=(z AND 56)/8:y=z AND 7:x
2=(z AND 3584)/512:y2=(z AND 448)/64:rh=
SGN(x2-x):rv=SGN(y2-y):RETURN
3900 :
3901 REM  ***
3902 REM  *** Spielstufe aendern
3903 REM  ***
3904 :
3910 CLS#1:PRINT#1,CHR$(7);CHR$(10);"Neu
e Spielstufe:  0 1 2 3":x=st
3920 GOSUB 6060:WHILE NOT te
3930 LOCATE#1,19+2*x,2:PAPER#1,3:PEN#1,1
:PRINT#1,HEX$(x)
3940 GOSUB 6000:IF NOT(ta OR tr OR tu OR
 tl OR te) THEN 3970
3950 LOCATE#1,19+2*x,2:PAPER#1,0:PEN#1,3
:PRINT#1,HEX$(x)
3960 x=(x+tl-tr) AND 3:y=(y+ta-tu) AND 3
3970 WEND:CLS#1:PEN#1,3:PAPER#1,0:PRINT#
1,CHR$(7);CHR$(10);"O.k.":LOCATE 26,1:PA
PER 0:PEN 3:PRINT HEX$(x)
3980 st=x:x=-1:y=6:GOTO 3020
4000 :
4001 REM %%%%%%%%%%%%%%%%%%
4002 REM % Computer Zug %
4003 REM %%%%%%%%%%%%%%%%%%
4004 :
4010 CLS#1:PRINT#1,CHR$(10);"Ich ";CHR$(
255);"berlege."
4020 IF sc=1 THEN x=0 ELSE x=3
4030 z=&A300:FOR i=0 TO 7:FOR j=0 TO 7
4040 POKE z,(f(j,i) XOR x)*SGN(f(j,i))
4050 z=z+1:NEXT j,i
4060 POKE &A006,st+1:CALL &9000
4070 z=PEEK(&A009):IF z=0 THEN z=512:RET
URN
4080 r=INT(RND*z)*2
4090 z=PEEK(&A200+r)+256*PEEK(&A201+r)
4100 RETURN
5000 :
5001 REM %%%%%%%%
5002 REM % Ende %
5003 REM %%%%%%%%
5004 :
5010 CLS#1:PEN#1,1:PRINT#1,CHR$(7);"Spie
lende";:PEN#1,3:IF fp=2 THEN PRINT#1," (
kein Zug mehr m";CHR$(254);"glich)"
5020 f=s(1):s(1)=0:s(2)=0
5030 FOR i=0 TO 7:FOR j=0 TO 7:s(f(j,i))
=s(f(j,i))+1:NEXT j,i
5040 LOCATE#1,1,2:PRINT#1,"Ergebnis: ";:
PEN#1,f:PRINT#1,s(1);:PEN#1,3:PRINT#1,":
";:PEN#1,f XOR 3:PRINT#1,s(2);
5050 IF sm=2 THEN 5100
5060 LOCATE#1,22,2:PEN#1,3
5070 IF s(1)=s(2) THEN PRINT#1,"Unentsch
ieden.";
5080 IF s(sc)>s(ss) THEN PRINT#1,"Ich ha
be gewonnen.";
5090 IF s(sc)<s(ss) THEN PRINT#1,"Du has
t gewonnen.";
5100 GOSUB 6060:WHILE NOT te
5110 GOSUB 6000:WEND
5120 GOTO 30
6000 :
6001 REM %%%%%%%%%%%%%%%%%%%%%%
6002 REM % Tastaturabfrage %
6003 REM %%%%%%%%%%%%%%%%%%%%%%
6004 :
6010 t$=INKEY$
6020 ta=(t$=CHR$(11)):tu=(t$=CHR$(10))
6030 tl=(t$=CHR$(8)):tr=(t$=CHR$(9))
6040 te=(t$="*")
6050 RETURN
6060 ta=0:tu=0:tl=0:tr=0:te=0
6070 WHILE INKEY$<>"":WEND
6080 RETURN
6100 :
6101 REM %%%%%%%%%%%%%%%%%%%%%%%%%
6102 REM % Spielstein setzen %
6103 REM %%%%%%%%%%%%%%%%%%%%%%%%%
6104 :
6110 ORIGIN 200+32*x,326-32*y
6120 FOR l=-5.5 TO 5.5
6130 sl=CINT(SQR(36-l^2))*2
6140 MOVE -sl,-INT(l)*2:DRAWR 2*sl-2,0,f
6150 NEXT:RETURN
```

```
7000 :
7001 REM %%%%%%%%%
7002 REM % Error %
7003 REM %%%%%%%%%
7004 :
7010 MODE 1:INK 0,0:INK 1,13:INK 2,6:PAP
ER 0:BORDER 1:CLS
7020 ORIGIN 0,0,0,640,0,400:CLG 0
7030 PLOT -10,0,1:MOVE 64,288
7040 PRINT CHR$(5);CHR$(162);
7050 LOCATE 17,3:PEN 2:PRINT"Error."
7060 LOCATE 2,6:PEN 1:PRINT"Bitte Comput
er mit <Ctrl><Shift><Esc>"
7070 PRINT CHR$(10);" zurucksetzen und P
rogramm erneut":PRINT CHR$(10);" laden !
"
7080 CALL &BB00:LOCATE 1,15
7090 END
8000 :
8001 REM %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8002 REM % Maschinenprogramm einlesen %
8003 REM %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
8004 :
8010 RESTORE 10210:p=0
8020 FOR i=0 TO 57:FOR j=0 TO 15
8030 READ a$:a=VAL("&"+a$)
8040 p=p+a:POKE &9000+16*i+j,a
8050 NEXT
8060 READ l:IF p<>l THEN 8100
8070 NEXT
8080 RETURN
8100 CLS:PRINT"DATA-Error in Zeile";
8110 PRINT 10210+i*10
8120 PRINT"Bitte korrigieren !"
8130 PRINT:CALL &BB00:CLEAR
8140 END
10000 :
10001 REM %%%%%%%%
10002 REM % Daten %
10003 REM %%%%%%%%
10004 :
10010 REM  ***
10011 REM  *** Zugrichtungen
10012 REM  ***
10013 :
10020 DATA 0,-1,0,1,-1,0,1,0,-1,-1,1,-1,
1,1,-1,1
10100 :
10101 REM  ***
10102 REM  *** Zeichenumdefinitionen
10103 REM  ***
10104 :
10110 DATA F0,18,3C,7E,7E,18,18,18,00,F1
,18,18,18,7E,7E,3C,18,00,F2,00,30,70,FE,
FE,70,30,00,F3,00,18,1C,FE,FE,1C,18,00,F
4,F8,F8,F0,F8,DC,0E,06,00,F5,3E,3E,1E,3E
,76,E0,C0,00,F6,C0,E0,76,3E,1E,3E,3E,00,
F7,06,0E,DC,F8,F0,F8,F8,00
10120 DATA FD,66,00,78,0C,7C,CC,76,00,FE
,66,00,3C,66,66,66,3C,00,FF,66,00,66,66,
66,66,3E,00
10200 :
10201 REM  ***
10202 REM  *** Maschinenprogramm
10203 REM  ***
10204 :
10210 DATA CD,89,92,AF,32,09,A0,3C,32,00
,A0,32,05,A0,3C,32,1477
10220 DATA 01,A0,FD,21,00,A0,F3,CD,48,91
,FB,C9,DD,21,0A,A0,3625
10230 DATA 11,06,00,FD,46,05,DD,19,10,FC
,C9,FD,5E,05,16,00,5065
10240 DATA 06,06,A7,CB,13,CB,12,10,FA,21
,00,A3,19,C9,78,82,6625
10250 DATA 57,79,83,5F,79,87,87,87,80,85
,6F,C9,F5,C5,D5,CD,9018
10260 DATA 2B,90,D1,D5,7B,07,07,07,82,85
,6F,D1,C1,F1,C9,15,11010
10270 DATA 1D,01,01,00,18,13,14,1D,01,08
,00,18,0C,15,1C,01,11228
10280 DATA 01,00,18,05,15,1D,01,08,00,CD
,4C,90,FD,36,04,FC,12305
10290 DATA 1E,03,7E,B7,C8,09,1D,20,F9,FD
,36,04,01,C9,15,1D,13729
10300 DATA 18,0A,14,1D,18,06,14,1C,18,02
,15,1C,CD,4C,90,FD,14643
10310 DATA 36,04,E7,7E,B7,C8,FD,36,04,01
,C9,15,18,07,14,18,16050
10320 DATA 04,1D,18,01,1C,CD,4C,90,FD,36
,04,01,A6,C8,FD,36,17546
10330 DATA 04,0A,C9,FD,35,05,CD,2B,90,FD
,34,05,E5,11,40,00,19084
10340 DATA 19,EB,E1,01,40,00,ED,B0,C9,3A
,05,A0,FD,BE,06,30,20968
10350 DATA 0A,DD,6E,06,DD,66,07,22,07,A0
,C9,CD,2B,90,E5,21,22701
10360 DATA FF,A0,01,00,80,1E,00,16,00,E3
,7E,23,E3,23,B7,28,24170
10370 DATA 35,32,02,A0,7E,CB,7F,28,19,CB
,BF,C5,D5,E5,C6,90,26331
10380 DATA 6F,26,93,6E,26,90,3A,02,A0,CD
,F8,FF,3A,04,A0,E1,28294
10390 DATA D1,C1,FD,35,02,20,02,ED,44,E5
,26,00,6F,CB,7F,28,30091
10400 DATA 01,25,09,44,4D,E1,14,CB,5A,28
,BE,1C,CB,5B,28,B7,31596
10410 DATA FD,71,07,FD,70,08,E1,C9,CD,C3
,90,3A,00,A0,32,01,33581
10420 DATA A0,EE,03,32,00,A0,CD,1C,90,3E
,FE,FD,86,01,DD,77,35613
10430 DATA 00,DD,77,01,DD,36,05,00,CD,2B
,90,2B,E5,DD,36,02,37175
10440 DATA 00,DD,36,03,00,E1,23,E5,7E,B7
,C2,68,92,DD,36,04,38974
10450 DATA 00,DD,56,03,DD,5E,02,FD,36,04
,00,DD,7E,04,07,C6,40468
10460 DATA 80,6F,26,93,46,23,4E,E1,E5,CD
,3E,90,7A,B3,CB,5F,42539
10470 DATA C2,5E,92,7E,B7,CA,5E,92,FD,BE
,00,20,06,FD,36,04,44516
10480 DATA 01,18,E6,3A,04,A0,B7,CA,5E,92
,DD,56,03,DD,5E,02,46245
10490 DATA E1,E5,3A,01,A0,77,CD,3E,90,3A
,00,A0,BE,20,05,EE,48131
10500 DATA 03,77,18,F2,3A,05,A0,FD,BE,06
,30,1E,FD,34,05,CD,49784
10510 DATA 48,91,DD,7E,05,B7,20,0C,CD,EB
,90,2A,07,A0,DD,75,51711
10520 DATA 00,DD,74,01,FD,35,05,CD,1C,90
,CD,D9,90,CD,C3,90,53847
10530 DATA DD,5E,00,DD,56,01,2A,07,A0,CD
,B8,FF,FD,CB,05,46,55854
10540 DATA 28,01,3F,38,0F,DD,75,00,DD,74
,01,3A,05,A0,3D,20,57021
10550 DATA 39,32,09,A0,3A,05,A0,3D,20,30
,DD,5E,00,DD,56,01,58284
10560 DATA CD,B8,FF,20,25,FD,6E,09,FD,34
,09,16,00,DD,5E,04,60024
10570 DATA A7,CB,1B,CB,1B,CB,1B,CB,12,DD
,7E,03,07,07,07,DD,61694
10580 DATA B6,02,B3,CB,05,26,A2,77,23,72
,DD,36,05,01,DD,34,63287
10590 DATA 04,DD,CB,04,5E,CA,81,91,DD,34
,03,DD,CB,03,5E,CA,65288
10600 DATA 75,91,DD,34,02,DD,CB,02,5E,CA
,71,91,3A,00,A0,32,67073
10610 DATA 01,A0,EE,03,32,00,A0,E1,C9,CD
,00,B9,21,12,A3,0E,68729
10620 DATA 04,06,04,7E,23,B7,28,1D,10,F9
,23,23,23,23,0D,20,69606
10630 DATA F0,3E,FC,32,7F,90,3E,E7,32,A2
,90,3E,0A,32,C1,90,71589
10640 DATA 21,40,93,18,12,3E,F6,32,7F,90
,3E,9C,32,A2,90,3E,73140
10650 DATA 28,32,C1,90,21,00,93,11,00,A1
,01,40,00,ED,B0,C9,74604
```

```
10660 DATA 00,00,00,00,00,00,00,00,00,00
,00,00,00,00,00,00,74604
10670 DATA 00,00,00,00,00,00,00,00,00,00
,00,00,00,00,00,00,74604
10680 DATA 00,00,00,00,00,00,00,00,00,00
,00,00,00,00,00,00,74604
10690 DATA 64,88,28,28,28,28,89,64,8A,84
,80,80,80,80,85,8A,76290
10700 DATA 28,83,05,02,02,05,81,28,28,83
,02,01,01,02,81,28,76990
10710 DATA 28,83,02,01,01,02,81,28,28,83
,05,02,02,05,81,28,77690
10720 DATA 8B,87,82,82,82,82,86,8B,64,88
,28,28,28,28,89,64,79390
10730 DATA 19,88,0A,0A,0A,0A,89,19,8A,84
,80,80,80,80,85,8A,80806
10740 DATA 0A,83,02,02,02,02,81,0A,0A,83
,02,01,01,02,81,0A,81380
10750 DATA 0A,83,02,01,01,02,81,0A,0A,83
,02,02,02,02,81,0A,81954
10760 DATA 8B,87,82,82,82,82,86,8B,19,88
,0A,0A,0A,0A,89,19,83384
10770 DATA 00,FF,00,01,FF,00,01,00,FF,FF
,01,FF,01,01,FF,01,84920
10780 DATA 5F,66,6D,74,8E,92,96,9A,AB,AE
,B1,B4,00,00,00,00,86636
10790 :
```

Directory-Druck

Wie der Name des Programms es schon andeutet, kann man das Inhaltsverzeichnis einer Diskette ausdrucken lassen und das in einer Form, die jedes Geo-Dreieck und jede Schneidemaschine überflüssig macht (Schneider CPC).

Man hat die Wahl zwischen einer „Catcopy" (sortierter Ausdruck mit Größenangabe jeder Datei) und einer „Dircopy" (unsortierter Ausdruck ohne Größenangabe). Wer schon eine gewisse Anzahl Disketten sein eigen nennen kann, wird aus leidiger Erfahrung die Suche nach der richtigen Diskette und der richtigen Datei nur zu gut kennen. Welche Wohltat, wenn man nur noch auf die Diskettenhülle zu schauen braucht.

Natürlich braucht man dazu einen Drucker. Die Steuercodes und die Texthardcopy (Zeilen 250 bis 360) sind auf den NLQ 401 abgestimmt. Epson-kompatible Drucker dürften den Dienst aber auch nicht verweigern. Der Rest des Programmes erklärt sich aus den REM- und PRINT-Anweisungen.

Otto Hubenschmid

```
10 'Titelbild
20 MODE 0:INK 1,26:INK 0,0:PAPER 1:PEN 0
:BORDER 0:CLS
30 LOCATE 5,5:PRINT"C D C O P Y"
40 LOCATE 8,10:PRINT"C P C"
50 LOCATE 4,13:PRINT"6128 664 464"
60 LOCATE 9,17:PRINT"von"
70 LOCATE 3,21:PRINT"OTTO HUBENSCHMID"
80 FOR n=1 TO 3000:a$=INKEY$:IF a$="" TH
EN NEXT n
90 '
100 '
110 'Erklaerung
120 MODE 2:CLS:BORDER 26:LOCATE 1,2:PRIN
T"CDCOPY":LOCATE 1,5
130 PRINT"gibt das Inhaltsverzeichnis ei
ner Diskette auf dem Drucker aus,"
140 PRINT"entweder mit 'cat' oder mit '!
dir'."
150 LOCATE 15,8:PRINT"[ Auswahl nachher
mit (C) oder (D) ].":LOCATE 1,12
160 PRINT"'cat' bedeutet sortierter Aus
druck mit Groessenangabe jeder Datei."
170 LOCATE 1,16
180 PRINT"'!dir' bedeutet unsortierter A
usdruck ohne Groessenangabe."
190 LOCATE 15,24
200 PRINT CHR$(24);" Drucker einschalten
 und eine Taste druecken! ";CHR$(24)
210 CALL &BB06
220 '
230 '
240 'Texthardcopy - Vorbereitung
250 PRINT#8,CHR$(27);CHR$(73);CHR$(1);CH
R$(27);CHR$(55);CHR$(15);
260 PRINT#8,CHR$(27);CHR$(51);CHR$(13);C
HR$(27);CHR$(83);CHR$(0);
270 MEMORY &A100-1
280 FOR i=-24320 TO -24222:READ byte:POK
E i,byte:s=s+byte:NEXT
290 DATA &cd,&78,&bb,&22,&64,&a1,&cd,&11
,&bc,&17,&32,&63,&a1,&21,&01,&01
300 DATA &22,&66,&a1,&3a,&63,&a1,&47,&0e
,&14,&c5,&e5,&cd,&75,&bb,&e1,&cd
310 DATA &60,&bb,&c1,&38,&02,&3e,&20,&cd
,&58,&a1,&e5,&c5,&3e,&42,&cd,&1e
320 DATA &bb,&c1,&e1,&20,&1c,&24,&0d,&20
,&e0,&10,&dc,&3e,&0d,&cd,&58,&a1
330 DATA &3e,&0a,&cd,&58,&a1,&2a,&66,&a1
,&2c,&22,&66,&a1,&7d,&fe,&1a,&20
340 DATA &c2,&2a,&64,&a1,&cd,&75,&bb,&c9
,&c5,&cd,&2e,&bd,&38,&fb,&cd,&31
350 DATA &bd,&c1,&c9
360 IF s<>11873 THEN MODE 2:CLS:PRINT"Fe
hler in Datazeilen":END
370 '
380 '
390 'Eingaben
400 MODE 2:CLS:BORDER 26:PRINT"O.K."
410 LOCATE 1,5:PRINT"Diskette einlegen!"
420 LOCATE 1,10:LINE INPUT"Disketten-Bez
. eingeben : ",bez$
430 LOCATE 1,14:LINE INPUT"Diskettenseit
e eingeben : ",seite$
440 bez$=LEFT$(bez$,20):seite$=UPPER$(LE
FT$(seite$,1))
450 LOCATE 1,20:PRINT"(C)ATCOPY oder (D)
IRCOPY ?"
460 a$=INKEY$:IF a$="" THEN 460
470 wahl$=UPPER$(a$):IF wahl$<>"C" AND w
ahl$<>"D" THEN 460
480 GOSUB 620
490 '
500 '
510 'weitere CDCOPIES
520 WINDOW#3,65,80,1,25:PAPER#3,0:PEN#3,
1:CLS#3
530 LOCATE#3,1,7:PRINT#3,"Weitere":PRINT
#3,"Diskette (J)"
540 LOCATE#3,1,11:PRINT#3,"Beenden  (B)"
550 a$=INKEY$:IF a$="" THEN 550
560 was$=UPPER$(a$):IF was$="J" THEN PAP
ER 1:PEN 0:GOTO 400
570 IF was$="B" THEN MODE 2:END
580 GOTO 550
590 '
600 '
610 'Unterroutine CDCOPY
620 MODE 2:BORDER 0:PAPER 0:PEN 1
630 PRINT STRING$(61,"-"):PRINT#8,STRING
```

```
$(61,"-")
640 WINDOW#2,3,60,1,25:PAPER#2,0:PEN#2,1
:CLS#2:CLS
650 WINDOW SWAP 0,2:IF wahl$="C" THEN CA
T ELSE |DIR
660 WINDOW SWAP 2,0
670 FOR y=1 TO 25:LOCATE 1,y:PRINT"|":LO
CATE 61,y:PRINT"|":NEXT y
680 LOCATE 1,25:PRINT STRING$(61,"-")
690 LOCATE 18,23:PRINT STRING$(44,"-"):L
OCATE 18,24:PRINT"|"SPC(1)bez$;
700 IF wahl$="C" THEN PRINT TAB(44);"C A
 T"; ELSE PRINT TAB(44);"D I R";
710 PRINT TAB(53);"Seite ";seite$
720 CALL &A100:RETURN
```

```
Drive A: user  0

CDCOPY1 .BAS   3K   CDCOPY4 .BAS    3K   SDBAENK3.BAS  16K
CDCOPY2 .BAS   3K   SDBAENK1.BAS   16K   SDBAENK4.BAS  16K
CDCOPY3 .BAS   3K   SDBAENK2.BAS   16K

102K free

                | Musterausdruck          C A T    Seite A |
```

```
Drive A: user  0

SDBAENK5.BAS   SDBAENK6.BAS   CDCOPY5 .BAS   CDCOPY6 .BAS
CDCOPY7 .BAS   SDBAENK7.BAS   SDBAENK8.BAS   CDCOPY8 .BAS

102K free

                | Musterausdruck          D I R    Seite B |
```

Musterausdruck

Verbesserte TRACE-Funktion

Die Verwendung der TRACE-Funktion des CPC-BASIC füllt den Bildschirm sehr schnell mit Zeilennummern, von einem geplanten Bildaufbau bleibt nicht mehr viel übrig. Hier sind zwei Maschinenprogramme, die dies ändern (CPC 464).

Das erste Maschinenprogramm verändert die TRACE-Funktion so, daß die Zeilennummern immer ins Window #7 ausgegeben werden. Durch geeignete Wahl der Window-Größe und entsprechende Positionierung auf dem Bildschirm kann eine Störung des Bildaufbaus weitgehend verhindert werden.

Listing 1 zeigt den BASIC-Loader für das Maschinenprogramm. Er bringt das Maschinenprogramm unterhalb von HIMEN in den Speicher, setzt HIMEN entsprechend herab und verbindet das Maschinenprogramm mit dem BASIC-Interpreter.

Wenn die drei unterstrichenen Null-Bytes in Zeile 1080 durch CD,18,BB ersetzt werden, wartet das Maschinenprogramm nach der Ausgabe jeder Zeilennummer auf einen Tastendruck. Erst nach dem Drücken einer beliebigen Taste (außer der ESC-Taste) läuft das Programm weiter. Auf diese Weise wird eine Einzelschritt-Funktion realisiert.

Listing 2 zeigt den BASIC-Loader für das zweite Maschinenprogramm. Auch hier erfolgt die Ausgabe ins Window #7. Es wird jedoch nicht nur die Zeilennummer ausgegeben, sondern die gesamte BASIC-Zeile. Das Listing läuft sozusagen auf dem Bildschirm mit.

Wenn die drei unterstrichenen Null-Bytes in Zeile 1080 durch CD,18,BB ersetzt werden, wartet auch dieses Maschinenprogramm nach der Ausgabe einer BASIC-Zeile auf einen Tastendruck. Erst nach dem Drücken einer beliebigen Taste (außer ESC) läuft das Programm weiter. Auf diese Weise läßt sich auch hier eine Einzelschritt-Funktion realisieren.

Gerd Kluge

```
10 REM Listing 1
20 REM CPC 464: Verbesserte Trac
e-Funktion
30 REM (Zeilennummern in Window
#7)
40 )
100 MEMORY HIMEM-100:init=HIMEM+
1
110 start=HIMEM+13:summe=0
120 FOR adr=init TO init+99
130 READ b$:byte=VAL("&"+b$)
140 POKE adr,byte:summe=summe+by
te
150 NEXT adr
160 IF summe<>12819

                    THEN PRINT "
DATA-Fehler!":STOP
170 hi=INT(start/256):lo=start-2
56*hi
180 POKE init+1,lo:POKE init+2,h
i
190 CALL init
1000 DATA 21,00,00,3E,C3,22,5B,B
B,32,5A
1010 DATA BB,C9,F5,DD,E1,E3,3E,C
3,BC,20
1020 DATA 03,3E,74,BD,E3,20,17,E
5,D5,21
1030 DATA 00,00,39,11,10,00,19,7
E,FE,A6
1040 DATA 20,04,23,7E,FE,DD,D1,2
8,07,E1
1050 DATA DD,E5,F1,CF,00,94,3A,0
C,B2,F5
1060 DATA 3E,07,CD,B4,BB,3E,5B,C
D,56,C3
1070 DATA 2A,36,AE,7E,23,66,6F,C
D,79,EE
1080 DATA 3E,5D,CD,56,C3,00,00,0
0,F1,CD
1090 DATA B4,BB,E1,F1,F1,F1,F1,F
1,F1,C9
```

Listing 1

```
10 REM Listing 2
20 REM CPC 464: Verbesserte Trac
e-Funktion
30 REM (Basic-Zeilen in Window #
7)
40 REM
100 MEMORY HIMEM-99:init=HIMEM+1
110 start=HIMEM+13:summe=0
120 FOR adr=init TO init+98
130 READ b$:byte=VAL("&"+b$)
140 POKE adr,byte:summe=summe+by
te
150 NEXT adr
160 IF summe<>12817

                    THEN PRINT "
DATA-Fehler!":STOP
170 hi=INT(start/256):lo=start-2
56*hi
180 POKE init+1,lo:POKE init+2,h
i
190 CALL init
1000 DATA 21,00,00,3E,C3,22,5B,B
B,32,5A
1010 DATA BB,C9,F5,DD,E1,E3,3E,C
3,BC,20
1020 DATA 03,3E,74,BD,E3,20,17,E
5,D5,21
1030 DATA 00,00,39,11,10,00,19,7
E,FE,A6
1040 DATA 20,04,23,7E,FE,DD,D1,2
8,07,E1
1050 DATA DD,E5,F1,CF,00,94,3A,0
C,B2,F5
1060 DATA 3E,07,CD,B4,BB,CD,4E,C
3,2A,36
1070 DATA AE,2B,2B,CD,63,E1,CD,4
5,E1,23
1080 DATA 7E,B7,20,F8,00,00,00,F
1,CD,B4
1090 DATA BB,E1,F1,F1,F1,F1,F1,F
1,C9
```

Listing 2

Formatieren unter BASIC

Dieses Programm für den Schneider CPC 464 ermöglicht es, eine Diskette unter BASIC zu formatieren. Bisher gab es diese Möglichkeit nur unter CPM, obwohl die Routinen dafür im ROM schon vorgesehen sind. *Andreas Müller*

```
10 ' *** Formatierung unter BASIC ***
40 MODE 1
50 INK 0,14:INK 1,0:INK 2,3:INK 3,14,3:B
ORDER 14
60 SPEED INK 20,30
70 WINDOW#1,1,40,1,4:PAPER#1,2:PEN#1,0:C
LS#1
80 WINDOW#0,1,40,6,25
90 '
100 PRINT#1:PRINT#1,TAB(16)"FASTFORMAT"
110 PEN 1
120 PRINT"Bitte eine Diskette einlegen"
130 PRINT"und mit";:PEN 2:PRINT"[ENTER]
";:PEN 1:PRINT"bestaetigen..."
140 WHILE INKEY$<>CHR$(13):WEND
150 CLS
170 GOSUB 400
190 LOCATE 1,7:PRINT"System-Format
(1)"
200 LOCATE 1,9:PRINT"Data-Only-Format
(2)"
210 LOCATE 7,19:PEN 3:PRINT"Ihre Wahl: "
220 a$=INKEY$:IF a$="" THEN 220
230 IF a$="1" THEN y=&41:ze=7
240 IF a$="2" THEN y=&C1:ze=9 ELSE IF a$
<>"1" AND a$<>"2" THEN 220
260 LOCATE 7,19:PRINT STRING$(10,32)
270 ad=&8035
280   FOR a=1 TO 9
290 POKE ad,0:POKE ad+1,0:POKE ad+2,y:PO
KE ad+3,2
300 ad=ad+4
310 y=y+2:IF (y AND 15)=11 THEN y=y-9
320   NEXT
340 LOCATE 25,ze:PRINT CHR$(242)+CHR$(15
4)
350 CALL &8000
370 LOCATE 7,19:PRINT "Noch eine Diskett
e formatieren (j/n)"
380 CALL &BB06:IF INKEY(45)=0 THEN RUN E
LSE CALL 0
400 DATA &3E,&00,&32,&2F,&80,&3A,&2F,&80
,&57,&3A, 665
410 DATA &30,&80,&5F,&3A,&31,&80,&4F,&21
,&35,&80, 799
420 DATA &DF,&32,&80,&3A,&2F,&80,&FE,&27
,&C8,&3C, 1187
430 DATA &32,&2F,&80,&21,&35,&80,&06,&09
,&77,&23, 608
440 DATA &23,&23,&23,&10,&F9,&18,&D6,&27
,&00,&41, 712
450 DATA &52,&C6,&07, 287
460 dat=0:sz=0:dz=400
470 FOR adr=&8000  TO &8034
480   READ byte:dat=dat+1
490   sz=sz+byte
500   POKE adr,byte
510   IF dat < 10  AND adr < &8034  THEN
 550
520   READ chksum
530   IF chksum<>sz THEN PRINT "Fehler i
n Zeile :";dz
540   dz=dz+10:sz=0:dat=0
550 NEXT adr
560 RETURN
```

Tip des Monats
Printer-Spooler

Das Programm wird vom Desktop aus gestartet. Danach laufen alle Druckerausgaben über den Spooler. Schon nach wenigen Sekunden meldet sich der Computer zurück, während der Drucker weiterarbeitet (Atari ST).

Das BASIC-Programm erzeugt die Datei ,,SPOOLER.PRG", die das Maschinenprogramm des Druckerspoolers enthält. Man muß dazu die Größe des Druckerpuffers in KB eingeben. Ferner fragt das Programm, ob die Druckeranpassung benötigt wird. Beim Gebrauch des Spoolers mit 1st-Word ist dies nicht der Fall, da 1st-Word eine eigene Druckeranpassung besitzt. Das erzeugte Programm ,,SPOOLER.PRG" kann durch Doppelklick vom Desktop gestartet werden. Danach laufen alle Druckerausgaben über den Spooler. Beim Ausgeben von Texten meldet der Computer sich schon nach wenigen Sekunden zurück, während der Drucker weiterarbeitet. Um den Spooler nach dem Laden des Betriebssystems automatisch zu initialisieren, kann auf der TOS-Diskette ein ,,AUTO"-Ordner angelegt und ,,SPOOLER.PRG" dort hineinkopiert werden.

Der Spooler besitzt eine Druckeranpassung für den Drucker FX-80, so daß sich jetzt auch die deutschen Umlaute und das ,,ß" drukken lassen. Die Druckeranpassungstabelle (Zeilen 600 bis 650 im BASIC-Programm) läßt sich auch für jeden anderen Drucker ändern. Bei Matrixdruckern ist es sogar möglich, fehlende Zeichen mit Hilfe von Grafik darzustellen. Die Tabelle ist folgendermaßen aufgebaut:

1. Byte: ASCII-Code im Atari
2. Byte: Anzahl der zum Drucker zu sendenden Codes
3 – n. Byte: Zum Drucker zu sendende Codes, durch die der Atari-ASCII-Code ersetzt wird
n + 1. Byte: ASCII-Code im Atari oder 0 (Tabellenende)
n + 2. Byte: Anzahl der zum Drucker ...

Beispiel:
142,4,27,82,2,91,153,4,27,82,2,92,0

Diese kleine Tabelle ersetzt den Code 142(Ä) durch die vier Byte 27,82,2 (bei Epson FX-80 deutscher Zeichensatz) und 91 und den Code 153(ö) durch die vier Byte 27,82,2 und 92.

Die Programmlänge des Maschinenprogramms steht in Zeile 250 im fünften und sechsten Data-Byte (2 ∗ 256 + 100 = 612 Byte). Wenn die Druckeranpassungstabelle verlängert wird, muß die Programmlänge um die Anzahl der neuen Byte erhöht werden. Da die Anzahl der Byte, um die die Tabelle verlängert wird, nicht ungerade sein darf, muß man die Tabelle in diesem Fall anstatt mit einem mit zwei Nullbyte abschließen. Nach Änderungen in den Datas sollte übrigens Zeile 230 gelöscht werden. *Mark-Oliver Stehr*

```
100    print"Spooler mit Druckeranpassung
        Vs.1.0 (c)1986 by M.O.Stehr"
110    r=1:s=0:open "R",#1,"SPOOLER.PRG",
       16:field #1,16 as a$
120    b$="":for i=0 to 15:read a:s=s+a:g
       osub 140:b$=b$+chr$(a):next
130    lset a$=b$:put #1,r:r=r+1:goto 120
140    if a<256 then return
150    on a-255 goto 160,180,190,220
160    input" -> Groesse des Druckerpuffe
       rs in kb :",size
170    c=size*1024:a=int(c/65536):b=int((
       c-a*65536)/256):return
180    a=b:return
190    input" -> Mit Druckeranpassung (j/
       n) ";d$
200    a=1:if d$="n" then a=0
210    return
220    lset a$=b$:put #1,r:close #1
230    if s<>42124 then print"Fehler in D
       atas !"
240    end
250    DATA 96,26,0,0,2,100,0,0,0,0,0,0,0
       ,0,0,0
260    DATA 0,0,0,0,0,0,0,0,0,0,0,0,47,60
       ,0,0
270    DATA 0,138,63,60,0,45,63,60,0,5,78
       ,77,80,143,35,192
280    DATA 0,0,0,174,47,60,0,0,1,124,66,
       103,63,60,0,13
290    DATA 78,78,80,143,65,249,0,0,2,2,7
       4,24,103,0,0,10
300    DATA 66,128,16,24,209,192,96,242,3
       5,200,0,0,1,232,35,200
310    DATA 0,0,1,244,35,200,0,0,1,240,66
       ,185,0,0,1,248
320    DATA 209,249,0,0,1,252,84,136,35,2
       00,0,0,1,236,44,111
330    DATA 0,4,32,60,0,0,1,0,208,174,0,1
       2,208,174,0,20
340    DATA 208,174,0,28,208,185,0,0,1,25
       2,84,128,66,103,47,0
350    DATA 63,60,0,49,78,65,34,79,48,17,
       8,0,0,13,102,0
360    DATA 0,4,78,105,12,105,0,3,0,6,103
       ,0,0,18,12,105
370    DATA 0,8,0,6,103,0,0,178,78,249,25
       5,255,255,255,74,105
380    DATA 0,8,102,244,20,41,0,11,74,120
       ,4,238,103,0,0,62
390    DATA 74,121,0,0,2,0,103,0,0,52,65,
       249,0,0,2,2
400    DATA 16,24,103,0,0,40,180,0,103,0,
       0,10,66,128,16,24
410    DATA 209,192,96,236,16,24,20,24,72
       ,231,128,128,97,0,0,20
420    DATA 76,223,1,1,83,0,102,238,112,2
       55,78,115,97,0,0,4
430    DATA 78,115,32,57,0,0,1,252,176,18
       5,0,0,1,248,103,242
440    DATA 74,185,0,0,1,248,102,0,0,24,8
       ,57,0,0,0,255
450    DATA 250,1,102,0,0,12,16,2,97,0,0,
       146,112,255,78,117
460    DATA 32,121,0,0,1,244,16,194,82,18
       5,0,0,1,248,177,249
470    DATA 0,0,1,236,102,0,0,8,32,121,0,
       0,1,232,35,200
480    DATA 0,0,1,244,112,255,78,117,74,1
       05,0,8,102,0,255,74
490    DATA 32,57,0,0,1,252,176,185,0,0,1
       ,248,103,0,0,6
500    DATA 112,255,78,115,66,128,78,115,
       72,231,128,128,74,185,0,0
510    DATA 1,248,103,0,0,42,83,185,0,0,1
       ,248,32,121,0,0
520    DATA 1,240,16,24,177,249,0,0,1,236
       ,102,0,0,8,32,121
530    DATA 0,0,1,232,35,200,0,0,1,240,97
       ,0,0,16,76,223
540    DATA 1,1,19,252,0,254,0,255,250,17
       ,78,115,65,249,0,255
550    DATA 136,0,16,188,0,15,17,64,0,2,1
       6,188,0,14,16,16
560    DATA 8,128,0,5,17,64,0,2,8,192,0,5
       ,17,64,0,2
570    DATA 78,117,0,0,0,0,0,0,0,0,0,0,0,
       0,0,0
580    DATA 0,0,0,0,0,0,0,0,0,256,257,0,0
       ,258
590    REM Hier beginnt die Tabelle zur D
       ruckeranpassung
600    DATA 64,4,27,82,0,64,91,4,27,82,0,
       91,92,4,27,82,0,92
610    DATA 93,4,27,82,0,93,123,4,27,82,0
```



```
       ,123,124,4,27,82
620    DATA 0,124,125,4,27,82,0,125,126,4
       ,27,82,0,126,221,4
630    DATA 27,82,2,64,142,4,27,82,2,91,1
       53,4,27,82,2,92
640    DATA 154,4,27,82,2,93,132,4,27,82,
       2,123,148,4,27,82
650    DATA 2,124,129,4,27,82,2,125,158,4
       ,27,82,2,126,0
660    REM Tabellenende
670    DATA 0,0,0,0,2,18,6,16,20,6,6,6,6,
       8,28,76,10
680    DATA 56,6,8,32,8,6,10,6,18,6,22,10
       ,6,8,10,6
690    DATA 0,259
```

BASIC-Listing zum Printer-Spooler

```
METACOMCO ASSEMBLER V10.203                        Spooler mit Druckeranpassung Vs.1.0
  LOC        OBJECT          STMT              SOURCE STATEMENT
                                2
                                3 *            (c)1986 by M.O.Stehr
                                4
         =0001                  5 gemdos       equ 1
         =0031                  6 keep         equ $31
                                7
         =000E                  8 xbios        equ 14
         =000D                  9 mfpint       equ 13
                               10
         =000D                 11 bios         equ 13
         =0008                 12 bcostat      equ 8
         =0003                 13 bconout      equ 3
         =0005                 14 setexec      equ 5
                               15
0000'  2F3C 0000 008A          16 init         move.l #newbios,-(sp)       * bios-Vektor umbiegen
0006'  3F3C 002D               17              move.w #45,-(sp)
000A'  3F3C 0005               18              move.w #setexec,-(sp)
000E'  4E4D                    19              trap #bios
0010'  508F                    20              addq.l #8,sp
0012'  23C0 0000 00AE          21              move.l d0,oldbios+2
                               22
0018'  2F3C 0000 017C          23              move.l #prnint,-(sp)        * Busy-Interruptroutine
001E'  4267                    24              clr.w -(sp)
0020'  3F3C 000D               25              move.w #mfpint,-(sp)
0024'  4E4E                    26              trap #xbios                 * Interrupt freigeben
0026'  508F                    27              addq.l #8,sp
                               28
0028'  41F9 0000 001A          29              lea characters,a0           * Pufferstart=Tabellenende+1
002E'  4A18                    30 slp          tst.b (a0)+                 * ... also Tabellenende suchen
0030'  6700 000A               31              beq foundend
0034'  4280                    32              clr.l d0
0036'  1018                    33              move.b (a0)+,d0
0038'  D1C0                    34              add.l d0,a0
003A'  60F2                    35              bra slp
003C'  23C8 0000 0000          36 foundend     move.l a0,pstart
0042'  23C8 0000 000C          37              move.l a0,pin               * pin=pout=Pufferstart
0048'  23C8 0000 0008          38              move.l a0,pout
004E'  42B9 0000 0010          39              clr.l charcounter           * keine auszugebenden Zeichen
0054'  D1F9 0000 0014          40              add.l size,a0
005A'  5488                    41              addq.l #2,a0
005C'  23C8 0000 0004          42              move.l a0,pend              * Pufferende berechnen
                               43
0062'  2C6F 0004               44              move.l 4(sp),a6             * Programmlaenge =
0066'  203C 0000 0100          45              move.l #256,d0              *  base-page
006C'  D0AE 000C               46              add.l 12(a6),d0             *   + text section
0070'  D0AE 0014               47              add.l 20(a6),d0             *   + data section
0074'  D0AE 001C               48              add.l 28(a6),d0             *   + bss section
0078'  D0B9 0000 0014          49              add.l size,d0               *   + Puffer-Groesse
007E'  5480                    50              addq.l #2,d0
0080'  4267                    51              clr.w -(sp)
0082'  2F00                    52              move.l d0,-(sp)
0084'  3F3C 0031               53              move.w #keep,-(sp)          * nicht benutzten Speicher
0088'  4E41                    54              trap #gemdos                * wieder freigeben
                               55
008A'  224F                    56 newbios      move.l sp,a1                * Superviser-Stack
008C'  3011                    57              move.w (a1),d0              * Bei Aufruf vom User-Modus,
008E'  0800 000D               58              btst #13,d0
0092'  6600 0004               59              bne l0
0096'  4E69                    60              move.l usp,a1               * User-Stack benutzen
0098'  0C69 0003 0006          61 l0           cmp.w #bconout,6(a1)        * Zeichenausgabe
009E'  6700 0012               62              beq newbconout
00A2'  0C69 0008 0006          63              cmp.w #bcostat,6(a1)        * Statusabfrage
00A8'  6700 00B2               64              beq newbcostat
00AC'  4EF9 FFFF FFFF          65 oldbios      jmp -1
                               66
         =00FFFA01             67 mfp          equ $fffa01                 * Multi Function Peripheral
         =00FFFA11             68 mfp_isrb     equ mfp+16                  * Interrupt In - Service Register
                               69
         =04EE                 70 dumpflg      equ $4ee
                               71
                               72 * Neue Drucker-Routinen fuer Zeichenausgabe und Statusabfrage
                               73
00B2'  4A69 0008               74 newbconout   tst.w 8(a1)                 * Drucker ?
00B6'  66F4                    75              bne oldbios                 * nein -->
00B8'  1429 000B               76              move.b 11(a1),d2            * auszugebendes Zeichen
00BC'  4A78 04EE               77              tst.w dumpflg               * Bei Hardcopy keine Umwandlung !
00C0'  6700 003E               78              beq tabend
00C4'  4A79 0000 0018          79              tst.w convert               * Umwandlung ?
00CA'  6700 0034               80              beq tabend                  * nein -->
00CE'  41F9 0000 001A          81              lea characters,a0           * Tabelle fuer Druckeranpassung
00D4'  1018                    82 search       move.b (a0)+,d0
00D6'  6700 0028               83              beq tabend
00DA'  B400                    84              cmp.b d0,d2                 * Zeichen suchen ...
00DC'  6700 000A               85              beq found
00E0'  4280                    86              clr.l d0
00E2'  1018                    87              move.b (a0)+,d0
00E4'  D1C0                    88              add.l d0,a0
00E6'  60EC                    89              bra search
00E8'  1018                    90 found        move.b (a0)+,d0             * ... und durch d0 Bytes
00EA'  1418                    91 lp1          move.b (a0)+,d2             *     aus der Tabelle ersetzen
00EC'  48E7 8080               92              movem.l d0/a0,-(sp)
00F0'  6100 0014               93              bsr bconout2
00F4'  4CDF 0101               94              movem.l (sp)+,d0/a0
00F8'  5300                    95              subq.b #1,d0
00FA'  66EE                    96              bne lp1
00FC'  70FF                    97              moveq #-1,d0
00FE'  4E73                    98              rte
0100'  6100 0004               99 tabend       bsr bconout2                * Zeichen in d2 -> Puffer
0104'  4E73                   100              rte
                              101
0106'                         102 bconout2
0106'  2039 0000 0014         103 wait         move.l size,d0
010C'  B0B9 0000 0010         104              cmp.l charcounter,d0        * Puffer voll ?
0112'  67F2                   105              beq wait                    * ja: warten
0114'  4AB9 0000 0010         106              tst.l charcounter           * Puffer leer ?
011A'  6600 0018              107              bne l05                     * nein -->
011E'  0839 0000 00FF FA01    108              btst #0,mfp                 * busy ?
0126'  6600 000C              109              bne l05                     * ja ->
012A'  1002                   110              move.b d2,d0                * nein: Zeichen -> Drucker
012C'  6100 0092              111              bsr charout
0130'  70FF                   112              moveq #-1,d0
0132'  4E75                   113              rts
0134'  2079 0000 000C         114 l05          move.l pin,a0
013A'  10C2                   115              move.b d2,(a0)+             * Zeichen -> Puffer
013C'  52B9 0000 0010         116              addq.l #1,charcounter
0142'  B1F9 0000 0004         117              cmp.l pend,a0               * Pufferende erreicht ?
0148'  6600 0008              118              bne l1                      * nein -->
014C'  2079 0000 0000         119              move.l pstart,a0            * ja: pin = Pufferstart
0152'  23C8 0000 000C         120 l1           move.l a0,pin
0158'  70FF                   121              moveq #-1,d0
015A'  4E75                   122              rts
                              123
015C'  4A69 0008              124 newbcostat   tst.w 8(a1)                 * Drucker ?
```

```
0160'  6600 FF4A            125             bne oldbios              * nein -->
0164'  2039 0000 0014       126             move.l size,d0
016A'  B0B9 0000 0010       127             cmp.l charcounter,d0     * Puffer voll ?
0170'  6700 0006            128             beq 12                   * ja -->
0174'  70FF                 129             moveq #-1,d0             * nein: noch Platz im Puffer
0176'  4E73                 130             rte
0178'  4280                 131 12          clr.l d0
017A'  4E73                 132             rte
                            133
                            134 * Drucker-Interrupt-Routine
                            135
017C'  48E7 8080            136 prnint      movem.l d0/a0,-(sp)
0180'  4AB9 0000 0010       137             tst.l charcounter        * Noch Zeichen im Puffer ?
0186'  6700 002A            138             beq 14                   * nein -->
018A'  5399 0000 0010       139             subq.l #1,charcounter
0190'  2079 0000 0008       140             move.l pout,a0           * Ausgabe-Pointer
0196'  1018                 141             move.b (a0)+,d0          * Zeichen -> d0
0198'  B1F9 0000 0004       142             cmp.l pend,a0            * Pufferende erreicht ?
019E'  6600 0008            143             bne 13                   * nein -->
01A2'  2079 0000 0000       144             move.l pstart,a0         * ja: pout = Pufferstart
01A8'  23C8 0000 0008       145 13          move.l a0,pout
01AE'  6100 0010            146             bsr charout              * Ausgabe an Drucker
01B2'  4CDF 0101            147 14          movem.l (sp)+,d0/a0
01B6'  13FC 00FE 00FF FA11  148             move.b #%11111110,mfp_isrb  * bit 0 loeschen (SEI-Modus)
01BE'  4E73                 149             rte
                            150
       =00FF8800            151 soundgen    equ $ff8800              * Soundchip mit Ports
                            152
01C0'  41F9 00FF 8800       153 charout     lea soundgen,a0
01C6'  10BC 000F            154             move.b #15,(a0)
01CA'  1140 0002            155             move.b d0,2(a0)          * Zeichen -> Port B
01CE'  10BC 000E            156             move.b #14,(a0)
01D2'  1010                 157             move.b (a0),d0
01D4'  0880 0005            158             bclr #5,d0
01D8'  1140 0002            159             move.b d0,2(a0)
01DC'  08C0 0005            160             bset #5,d0               * Strobe erzeugen
01E0'  1140 0002            161             move.b d0,2(a0)
01E4'  4E75                 162             rts
                            163
0000'                       164             data
                            165
0000'  =0004                166 pstart      ds.l 1                   * Pufferstartadresse

0004'  =0004                167 pend        ds.l 1                   * Pufferendadresse
0008'  =0004                168 pout        ds.l 1                   * Ausgabe-Pointer
000C'  =0004                169 pin         ds.l 1                   * Eingabe-Pointer
0010'  =0004                170 charcounter ds.l 1                   * Anzahl der Zeichen im Puffer
                            171
0014'  0000 5000            172 size        dc.l 20*1024             * Groesse des Puffers (in Bytes)
0018'  0001                 173 convert     dc.w 1                   * 0 = keine Codeumwandlung
                            174 * Dieses Flag sollte beim Gebrauch mit 1st-Word auf 0 gesetzt werden,
                            175 * da 1st-Word bereits eine eingebaute Druckeranpassung besitzt.
                            176
                            177 *               1. 2. 3. 4. 5. 6. Byte
001A'  4004 1B52 0040       178 characters  dc.b '@',4,27,82,0,64    * Tabelle zur Drucker-Anpassung
0020'  5B04 1B52 005B       179             dc.b '[',4,27,82,0,91
0026'  5C04 1B52 005C       180             dc.b '\',4,27,82,0,92    * 1.Byte:   Atari-ASCII-Code
002C'  5D04 1B52 005D       181             dc.b ']',4,27,82,0,93    * 2.Byte:   Anzahl der zum Drucker
0032'  7B04 1B52 007B       182             dc.b '{',4,27,82,0,123   *           zu sendenden Bytes
0038'  7C04 1B52 007C       183             dc.b '|',4,27,82,0,124   * 3.-n.Byte: Zum Drucker zu
003E'  7D04 1B52 007D       184             dc.b '}',4,27,82,0,125   *           sendende Bytes
0044'  7E04 1B52 007E       185             dc.b '~',4,27,82,0,126
004A'  DD04 1B52 0240       186             dc.b ' ',4,27,82,2,64    * Hier EPSON FX-80:
0050'  8E04 1B52 025B       187             dc.b ' ',4,27,82,2,91
k. Zeichensatz
0056'  9704 1B52 025C       188             dc.b ' ',4,27,82,2,92    * 27,82,2 = deutscher Zeichensatz
005C'  9A04 1B52 025D       189             dc.b ' ',4,27,82,2,93
0062'  8404 1B52 027B       190             dc.b ' ',4,27,82,2,123   * Diese Tabelle kann je nach
0068'  9404 1B52 027C       191             dc.b ' ',4,27,82,2,124   * Drucker beliebig veraendert
006E'  8104 1B52 027D       192             dc.b ' ',4,27,82,2,125   * bzw. erweitert werden
0074'  9E04 1B52 027E       193             dc.b ' ',4,27,82,2,126
007A'  00                   194             dc.b 0                   * 0 = Ende
                            195
                            196             end

No errors found in this Assembly
```

Assembler-Listing zum Printer-Spooler

Easy-DATA

Beim Verfassen eigener Programme und beim Abtippen von Listings ist es oft nötig, eine riesige Menge von in DATA-Zeilen stehenden Werten in den Computer einzugeben. Dieses Programm vereinfacht die Sache erheblich und hilft somit auch Zeit sparen (Atari XL).

Man gibt am Anfang die Zeilennummer ein, an der die erste DATA-Zeile stehen soll, und den gewünschten Zeilenabstand. Eine weitere Erleichterung besteht in der Umfunktionierung der Leertaste. Bei Gebrauch von Autokomma genügt es statt der Kommataste, die so hinterhältig neben dem Punkt liegt, auf die Leertaste zu drücken. Die fehlenden Kommas werden automatisch eingesetzt. Gleichzeitig wird jedes O, das sich in die Zahlenreihe geschmuggelt hat, wieder in eine 0 verwandelt. Die eingetippte Zeile wird unter der nächsten angezeigt. Entdeckt man dann einen Fehler, drückt man zur Returngleichzeitig die Option-Taste. Die Zeile kann dann korrigiert werden, bevor es weiter geht. Nach Beendigung der Eingaben drückt man bei der folgenden Zeile (ohne Eingabe) RETURN. Es besteht dann die Möglichkeit, das Programm zu löschen, so daß nur noch die DATA-Zeilen im Speicher sind. Achtung vor zu langen Eingaben. Sie führen nicht zum Abbruch des Programms, aber die überschüssigen Zahlen werden nicht berücksichtigt und fehlen am Schluß.

Mit POKE 731, 255 (Normalwert 0) kann man nun das, manchem lästige, Tastaturklicken ausschalten. Der Warnton vor einer zu lagen Zeile ertönt trotzdem noch. Hinweis: Die Pfeile im Listing entstehen durch ESC/Control und < beziehungsweise =.

Stefan Neumann

```
NY 1 DIM D$(150),K$(1)
WB 2 ? CHR$(125)
DQ 3 ? :? "Erste Zeilennummer";:INPUT Z
DU 4 IF Z<49 THEN ? "BELEGT":FOR W5=1 TO
   500:NEXT W5:GOTO 2
AY 5 ? :? "Zeilenabstand";:INPUT A
RN 6 Z=Z-A
EW 7 ? :? "Autokomma [J/N]";:INPUT K$
WH 8 ? CHR$(125)
RE 9 Z=Z+A
XO 10 POSITION 2,4
IZ 11 ? Z;" DATA ";:INPUT D$:LD=LEN(D$)
SO 12 IF D$="" AND PEEK(53279)<>3 THEN 30
LY 13 IF K$="J" AND PEEK(53279)<>3 THEN 2
   3
XP 14 IF PEEK(53279)=3 THEN GOTO 29
QL 15 POSITION 2,21:? Z;" DATA ";D$
LM 16 ? "CONT";:POSITION 2,18
HW 17 POKE 842,13
ZF 18 END
HM 19 POKE 842,12
XM 20 ? CHR$(125)
IG 21 POSITION 2,9:? Z;" DATA ";D$
BV 22 GOTO 9
KR 23 FOR Q=1 TO LD
BJ 24 IF D$(Q,Q)=" " THEN D$(Q,Q)=","
JL 25 IF D$(Q,Q)="O" THEN D$(Q,Q)="0"
LW 26 NEXT Q
IB 27 IF D$(LD,LD)="," THEN D$(LD,LD)=" "
TN 28 GOTO 14
KP 29 Z=Z-A:GOTO 10
XN 30 ? CHR$(125)
WU 31 ? :? "Soll >EINGABEHILFE< geloescht
   werden"
CJ 32 ? :? "[J/N] ";:INPUT K$
```

```
IO 33 IF K$="J" THEN 35
TM 34 GRAPHICS 0:END
ZE 35 ? :? "Soll wirkilch geloescht werde
   n [J/N]";:INPUT K$
LC 36 IF K$="J" THEN 38
UE 37 GOTO 34
PF 38 FOR T=0 TO 37
VD 39 ? "K↓":LIST T
ML 40 ? T:? "CONT"
UL 41 POSITION 0,0
ZP 42 POKE 842,13:END
HD 43 POKE 842,12
MV 44 NEXT T
JV 45 ? "K↓"
ET 46 FOR T=38 TO 48:? T:NEXT T
DB 47 ? "POKE 842,12:L."
H5 48 POSITION 0,0:POKE 842,13:END
```

REM-Killer

Löscht selbständig alle reinen Kommentar- und Programmzeilen, die mit dem REM-Statement beginnen. Ober- und Untergrenze für den Löschvorgang sind programmierbar (Atari XL).

Das Programm wird in tokenisierter Form untersucht. Ist das erste Token einer Zeile eine Null (REM), wird eine Prozedur aufgerufen, die die Zeile löscht.

Aufruf des Programms:
GOTO KILL (Variable, die im Hauptprogramm vereinbart werden sollte). *Georg Pohl*

```
RD 30 POKE 709,0:POKE 710,10:POKE 712,8
VF 40 POKE 729,20:POKE 752,1
BD 50 KILL=32310:MAX=32767:REM SOLLTE
   VEREINBART SEIN ...
TY 100 PRINT CHR$(125)
OP 105 FOR J=2 TO 38:PRINT "-";:NEXT J:PR
   INT
OV 107 PRINT "REM KILLER-Benutzer-Hinweis
   e ..."
OY 108 FOR J=2 TO 38:PRINT "-";:NEXT J:PR
   INT
NY 109 POSITION 2,10
QI 110 PRINT "Zum starten des REM KILLER
   geben Sie"
PV 120 PRINT "GOTO KILL ein. Danach wird
   jede Pro-"
HF 130 PRINT "gramm-Zeile geloescht, die
   mit REM"
SS 140 PRINT "beginnt ..."
ZC 32290 END
RU 32310 X=PEEK(136)+256*PEEK(137)
MV 32320 L=PEEK(X)+256*PEEK(X+1)
IB 32330 R=PEEK(X+4)
MU 32335 IF L>MAX THEN END
JL 32340 IF R>0 THEN X=X+PEEK(X+2):GOTO 3
   2320
HF 32360 PRINT CHR$(125)
FK 32370 POSITION 2,4:PRINT L
YK 32380 PRINT "CONT"
LS 32390 POSITION 2,0
OH 32400 POKE 842,13:STOP
CW 32410 POKE 842,12
AY 32420 GOTO 32320
YK 32430 END
```

Page-Lister

Ermöglicht ein Auslisten nach Bildschirmseiten. Auf Wunsch kann ein LIST-Stopp bei jeder durch REM eingeleiteten Programmzeile vereinbart werden (Atari XL).

Beim Listen wird permanent die Speicherstelle ROWCRS (dez. 84) abgefragt. Erreicht sie Werte größer 20, wird das Listen bis auf Tastendruck unterbrochen. Der Bildschirm wird gelöscht und ab der obersten Zeile weitergeLISTet.

Bei Zusatzvereinbarung (Stopp auch bei REM) werden auch die Tokens jedes Zeilenanfangs abgefragt. Wird hier eine Null gefunden (für REM), wird ebenfalls bis auf Tastendruck unterbrochen.

Aufruf des Programms:
GOTO PAGE (Variable, die im Hauptprogramm vereinbart sein sollte).
PAGE-LISTER selbst kann auf diese Weise geLISTet werden. *Georg Pohl*

Wichtig:
Bei Verwendung des Atari-Checksummers müssen alle Zeilennummern in den Programmen „REM-Killer" und „Page-Lister" unter 32 000 liegen.

```
QB 10 REM ┌────────────────────────────────┐
LQ 11 REM │                                │
DU 12 REM │ PROGRAMMIER- U. EDITIER-       │
MH 13 REM │ HILFE PAGE_LISTER              │
LW 14 REM │                                │
GG 15 REM │ LISTET QUELL-TEXT NACH         │
SQ 16 REM │ BILDSCHIRMSEITEN ORDER         │
HF 17 REM │ WAHLWEISE NACH REM-ZEILEN      │
QY 18 REM │ UND BILDSCHIRMSEITEN OHNE      │
YO 19 REM │ SCROLLING ...                  │
LP 20 REM │                                │
ZC 21 REM │                (C) GEORG POHL  │
UP 22 REM │                    APRIL 1986  │
LV 23 REM │                                │
QI 24 REM └────────────────────────────────┘
BJ 25 REM
BL 26 REM
BN 27 REM
XT 30 POKE 709,0:POKE 710,10:POKE 712,10
UF 40 POKE 729,20:POKE 730,2
QO 50 CLOSE #3:OPEN #3,4,0,"K:"
JV 60 PAGE=32010
YV 70 END
IT 3217 IF P>21 OR R=0 THEN POSITION 1,23
   :PRINT "TASTE DRUECKEN, M=ABBRUCH";:GE
   T #3,M1
BI 32000 REM
BN 32001 REM ┌───────────────────────────┐
RO 32002 REM │ START DER ROUTINE PAGE    │
CW 32003 REM └───────────────────────────┘
CC 32004 REM
DS 32010 POKE 82,1:PRINT CHR$(125):M1=0
XY 32015 CLOSE #3:OPEN #3,4,0,"K:"
ZX 32020 FOR J=1 TO 39:PRINT "-";:NEXT J
BL 32030 PRINT "  PAGE- UND REM-LISTER, A
   USWAHL ..."
AF 32040 FOR J=1 TO 39:PRINT "-";:NEXT J
TX 32050 POSITION 1,10
AD 32060 PRINT "(1) LISTEN MIT REM- U. SE
   ITENABSCHLUSS"
HC 32070 PRINT "ODER (2) NUR MIT SEITENAB
   SCHLUSS ? ";
GD 32080 TRAP 32070:GET #3,M:M=VAL(CHR$(M
   ))
CS 32090 REM
CX 32091 REM ┌───────────────────────────┐
SZ 32092 REM │ PAGE_LISTER AUSGABE ...   │
EG 32093 REM └───────────────────────────┘
DM 32094 REM
WJ 32100 POKE 82,2:PRINT CHR$(253);CHR$(1
   25)
NO 32110 MAX=32767:MIN=30
```

```
RS 32120 X=PEEK(136)+256*PEEK(137)
MT 32130 L=PEEK(X)+256*PEEK(X+1)
GC 32140 P=PEEK(84):R=1:IF M1=77 THEN END
DX 32150 IF M=1 THEN R=PEEK(X+4)
XH 32160 IF L>MAX THEN 32210
YC 32170 M1=0:IF P>20 OR R=0 AND L>30 THE
     N POSITION 1,23:PRINT "      TASTE DRUE
     CKEN. M=ABBRUCH";:GET #3,M1
BA 32175 IF M1=77 THEN GOTO 32210
AU 32177 IF P>20 OR R=0 AND L>30 THEN PRI
     NT CHR$(125):P=1
DI 32180 POSITION 4,P-1:LIST L
UV 32190 X=X+PEEK(X+2)
ZX 32200 GOTO 32130
ZZ 32210 POSITION 2,23:PRINT CHR$(156);"
         LISTING-ENDE ....";
OF 32220 POSITION 2,21:END
IK 32230 REM IN ZEILE 32170 UND 32177
         KANN DER BEFEHL "AND L>30" WEG-
         GELASSEN WERDEN !!!
```

Directoryloader

Übersichtliche und komfortable Auswahl von Diskettenprogrammen auf dem Atari XL/XE

Der Directoryloader (DIRLOAD) ist ein Programm zur komfortablen und übersichtlichen Auswahl von Diskettenprogrammen. Aus Gründen der Übersicht auf dem Bildschirm werden maximal 40 Files angezeigt. Bei Disketten mit mehr als 40 Files werden nur die ersten 40 Files angezeigt. DIRLOAD arbeitet mit jedem DOS zusammen und ist vollständig in BASIC geschrieben.

DIRLOAD beinhaltet die Möglichkeit, daß nach dem Einschalten des Computers automatisch DIRLOAD geladen wird. Voraussetzung dafür ist allerdings ein DOS.SYS beziehungsweise FMS-SYS File auf der Diskette. Mit DIRLOAD können Sie auch das DOS aufrufen. Sollte der DOS-Vektor (PEEK(10), PEEK(11)) auf den Selbsttest des Atari 600XL/800XL/130XE's zeigen, wird angezeigt, daß kein DOS ladbar ist.

Nach Start des Programms wird ein Menü angezeigt, in dem Sie vier Programmpunkte wählen können:

A Directory laden
B Disk installieren
C DOS laden
D Programmende

Mit Funktion wird Ihnen das Directory der Disk angezeigt (maximal 40 Files). Es wird in folgender Form angezeigt:
Nummer,Filename,Länge — zum Beispiel:
00 DIRLOAD 10 K oder 12 FMS.Sys 4 K
Um nun ein File zu laden, tippen Sie nur die Nummer ein, dann drücken Sie RETURN, bei einstelligen Zahlen muß immer eine „0" vorweggedrückt werden.

Dann beantworten Sie die Frage, ob Sie das File starten wollen. Wenn Sie hier N = Nein drücken, wird das File nur geladen = LOAD. Andernfalls wird das File gestartet = RUN ...
Mit „B" wird eine Kopie von DIRLOAD auf die Disk geschrieben. Sollte bereits ein File namens DIRLOAD auf der Disk sein, werden Sie darauf aufmerksam gemacht und gefragt, ob Sie dieses File überschreiben wollen. Beim Abspeichern wird automatisch ein Autorun.Sys-File auf die Diskette geschrieben, welches ein automatisches Laden von DIRLOAD nach dem Einschalten des Computers ermöglicht.
Mit „C" können Sie, soweit möglich, das DOS laden.
Mit „D" können Sie aus dem Programm „aussteigen.
Während das Directory angezeigt wird (Funktion A), können Sie mit der Taste „C" sich ansehen, wieviel Platz noch auf der Disk ist, wieviel belegt ist und auf wieviel Kilobyte Disk formatiert ist. Mit Taste „D" können Sie sich den Bild-

```
      ZE 0 REM * DIRLOAD - VERSION 6.0 *
ZL 1 REM * FUER ATARI'S MIT DISK *
HY 2 REM * LAENGE 10 KB.          *
HL 3 REM * AUTOR :                *
ZP 4 REM * SOENKE HORN            *
ET 5 REM * IN DE BANS 5C          *
SA 6 REM * 2100 HAMBURG 90        *
BO 7 REM * TEL. 040/7602737       *
LI 8 REM * AUS DARSTELLUNGSGRUENDEN *
       * WURDE EINE MAXIMALE FILE *
       *ZAHL VON 40 FILES GEWAEHLT*
CV 10 DIM A$(40*12),B$(12),C$(3),D$(155),
   E$(20),F$(80),FL(40),G$(20)
IJ 11 D$(1)=" ":D$(155)=" ":D$(2)=D$
WZ 15 G$="LADEN"
SX 20 ERROR=500:START=600:POKE 65,1
OL 30 A$(1)=" ":A$(40*12)=" ":A$(2)=A$
UL 50 GOTO START
EH 60 IF  NOT X2 AND RET THEN 100
ZX 70 RETURN
NN 90 GRAPHICS 0:POKE 710,202:POKE 16,64:
   POKE 53774,64:POKE 752,1:POKE 82,2:POK
   E 709,0:X1=0:IF A$(1,1)<>" " THEN X2=1
EC 91 ? CHR$(125):POSITION 2,10:? "SOLL E
   IN BEREITS GELADENES DIRECTORY  ANGEZE
   IGT WERDEN (J/N) ?";:G$="LADEN"
SP 92 CLOSE #1:TRAP 90:OPEN #1,4,0,"K:"
XZ 93 GET #1,K:IF K=74 THEN TRAP 100:GOTO
    220
WA 94 IF K=78 THEN 100
QZ 95 IF K=155 THEN X1=X1+1:RET=1
NZ 96 IF RET AND X1=1 AND  NOT (X2) THEN
   ? "NEIN !";:FOR J=1 TO 4:? CHR$(30);:N
   EXT J:RET=0:GOTO 93
HU 97 IF RET AND X1=1 AND X2 THEN ? "JA
   !";:FOR J=1 TO 4:? CHR$(30);:NEXT J:RE
   T=0:GOTO 93
JQ 98 IF X1=2 THEN RET=0:IF X2 THEN TRAP
   100:GOTO 220
IL 99 GOSUB 60:GOTO 93
NO 100 G$="LADEN":TRAP ERROR:CLOSE #1:POK
   E 710,33:POKE 709,62:A$(1)=" ":A$(40*1
   2)=" ":A$(2)=A$:AZ=0:B$="":C$="":D$=""
   :E$=""
YQ 110 OPEN #1,6,0,"D:*.*":POKE 559,0
LA 120 TRAP 220
WJ 130 FOR I=1 TO 64
ED 140 INPUT #1,D$:D$=D$(3)
BV 145 FL(I-1)=INT(VAL(D$(13,15))/8):IF F
   L(I-1)=0 THEN FL(I-1)=1
RQ 150 B$(1,8)=D$(1,8):B$(9,9)=".":B$(10,
   12)=D$(9,11):D$="":X=0
EM 160 FOR J=1 TO 12:IF B$(J,J)=" " THEN
   NEXT J:GOTO 200
SG 170 X=X+1:D$(X,X)=B$(J,J):NEXT J
MW 180 IF D$(2)="FREES.ECT" THEN 220
CJ 200 A$(1+12*AZ,1+12*AZ+12)=D$:D$="":AZ
   =AZ+1
FU 210 NEXT I
RI 220 CLOSE #1:POKE 82,0:? CHR$(125):POK
   E 752,1:POKE 710,40:POKE 709,2:AB=0:GO
   SUB 1350:IF A$(1,1)=" " THEN 100
QY 221 FOR J=0 TO AZ-1
BB 222 IF J<10 THEN C$(1)="0":C$(2)=STR$(
   J):C$(3)=" "
XV 224 IF J>9 THEN C$(1,2)=STR$(J):C$(3)=
   " "
BK 225 IF J<10 THEN C$(1)="0":C$(2)=STR$(
   J):C$(3)=" "
OU 230 IF J>20 THEN AB=20:POSITION AB,J-2
   1
KC 240 IF J<20 THEN AB=0:POSITION AB,J
QE 250 ? C$;A$(1+12*J,1+12*J+11)
VM 255 GOSUB 550
GO 260 NEXT J
```


schirminhalt ausdrucken lassen, als Drucker ist ein Atari 1029 vorgesehen, es kann aber auch ein anderer Drucker sein. Wollen Sie vorzeitig das Ausdrucken abbrechen, drücken Sie einfach Select.

Fehleingaben sind fast unmöglich, durch Fehleingaben beziehungsweise BREAK kann DIRLOAD nicht unterbrochen werden.

DIRLOAD läuft auf allen Ataris mit mindestens 32 K und einem Diskettenlaufwerk.

Es ergeben sich einige Unterschiede bei DOS III. Folgende Zeilen sind zu ändern:

145 FL(I-1)=INT(VAL(D$(13,15))/8):IF FL(I-1)=0 THEN FL(I-1)=1
180 IF D$(2)="FREEB.LOC" THEN 220
1360 INPUT#1, E$:DCP=INT(VAL(E$)/8):CLOSE #1:RETURN

Die Änderungen sind wegen der anderen DOS-Gestaltung bei DOS III notwendig. Zum Beispiel: DOS II: 028 FREE SECTORS →
DOS III: 24 FREE BLOCKS

Sönke Horn

```
ZD 265 POSITION 8,23:? "TIPPE ";
WP 266 IF AZ<10 THEN ? "0";AZ;
BE 267 IF AZ>9 THEN ? AZ;
EU 268 ? " FUER MENUE EIN.";
OY 270 POKE 82,0:POSITION 6,21:? "TIPPE D
    EINE FILE-NUMMER EIN :"
NE 275 REM TEXT  IN 270 INVERS SCHREIBEN
XM 280 CLOSE #1:OPEN #1,4,0,"K:"
AJ 281 E$="":E$=STR$(AZ)
QH 282 IF AZ<10 THEN E$(1)="0":E$(2,2)=ST
    R$(AZ):X=ASC(E$(2,2)):X=X+128:E$(2)=CH
    R$(X):REM "0" IN 282 INVERS
JP 283 IF AZ>9 THEN FOR J=1 TO 2:X=ASC(E$
    (J,J)):X=X+128:E$(J,J)=CHR$(X):NEXT J
SC 290 POSITION 18,22:? ">";E$;"<":X=0:NR
    =0:TRAP 290:REM > UND < IN 283  INVERS
OM 300 G$="EINGEBEN":POKE 559,34:GET #1,K
    :IF K=126 THEN 350
CR 301 IF K=68 THEN G$="AUSDRUCKEN":GOSUB
     20000:GOTO 300
UO 302 IF K=125 THEN ? CHR$(K):SETCOLOR 2
    ,0,0:SETCOLOR 1,0,10:G$="LADEN":GOTO 1
    00
KL 305 IF K=67 THEN GOSUB 1160:GOTO 270
IL 310 IF K=155 THEN 380
IW 311 IF K<48 OR K>57 THEN 300
TR 320 X=X+1:IF X>2 THEN X=2
LL 330 POSITION 18+X,22:? CHR$(K+128)
MK 340 GOTO 300
EM 350 IF X>0 THEN POSITION 18+X,22:? "█"
    :X=X-1:REM ?" " LEERZEICHEN INVERS
MO 360 GOTO 300
MQ 370 GOTO 300
IC 380 LOCATE 19,22,A:A=A-128:C$(1)=CHR$(
    A)
FI 390 LOCATE 20,22,A:A=A-128:C$(2)=CHR$(
    A)
UO 400 NR=VAL(C$):E$(1,2)="D:"
NN 401 IF NR=AZ THEN 600
OI 405 IF NR<0 OR NR>AZ-1 THEN GOTO 220
WH 410 E$(3)=A$(1+12*NR,1+12*NR+11)
US 420 FOR J=21 TO 23:POSITION 0,J:? "
                                     ";:NEX
    T J:POKE 764,255
BP 430 POSITION 2,21:? "WOLLEN SIE DAS FI
    LE ";E$(3):? "  STARTEN (J/N) ?"
UJ 435 TRAP 470:G$="EINGEBEN"
WU 440 GET #1,K:IF K=ASC("N") THEN TRAP 5
    00:LOAD E$
DK 450 IF K=ASC("J") THEN RUN E$
OR 460 GOTO 440
DS 470 IF PEEK(195)<>21 THEN 500
YI 480 POSITION 2,21:? "DIESES PROGRAMM I
    ST EIN ENTER-FILE ":? "  SOLL ES TROTZ
    DEM GELADEN WERDEN (J/N)"
XQ 490 CLOSE #1:OPEN #1,4,0,"K:"
NO 491 TRAP 480:GET #1,K:IF K=ASC("N") TH
    EN 600
NK 492 IF K=ASC("J") THEN TRAP 500:ENTER
    E$
SB 495 GOTO 491
CP 500 POKE 82,2:FOR I=20 TO 23:POSITION
    0,I:? "
      ";:NEXT I:POSITION 0,20
XE 501 ? "FEHLER BEIM ";G$;" IST AUFGETRE
    TEN!":G$="EINGABE"
OL 510 ? "  BITTE DRUECKEN SIE EINE TASTE
     :"
MX 520 POKE 764,255:CLOSE #1:OPEN #1,4,0,
    "K:":POKE 559,34
GO 530 TRAP 500:GET #1,K:GOTO 600
UG 550 E$="":POSITION AB+15,J-(21*(AB=20)
    )
VD 560 X=FL(J):IF X<10 THEN E$(1)="0":E$(
    2)=STR$(X)
XS 570 IF X>9 THEN E$(1)=STR$(X)
BO 580 E$(3)=" K":? E$:RETURN
QU 600 GRAPHICS 17:POKE 712,19:POKE 16,64
    :POKE 53774,64:POKE 559,0
HI 610 POSITION 0,2
RN 620 ? #6;"  DIRECTORY-LOADER ":REM TEX
    T IN 620,650,670 INVERS SCHREIBEN
KC 640 ? #6
KY 650 ? #6;"  VERSION 06.0/D-2":? #6
NX 660 ? #6;"--------------------"
IZ 670 ? #6;"        menue       "
ZJ 680 ? #6;"A directory laden   "
CD 690 ? #6;"B disk installieren "
BK 695 ? #6;"C dos laden         "
RJ 700 ? #6;"D programm verlassen"
RB 705 ? #6;"                    "
NO 710 ? #6;"--------------------"
LJ 720 ? #6;"COPYRIGHT (C) 01.86":REM 720
    ,730 TEXT INVERS
IE 730 ? #6;"  BY HORNSOFTWARE"
UL 735 POKE 764,255:POKE 559,34
CT 740 POKE 710,(20*(PEEK(20)>100)+40*(PE
    EK(20)<100)):IF PEEK(20)>200 THEN POKE
     20,0
IM 750 IF PEEK(764)=255 THEN 740
XX 760 IF PEEK(764)=63 THEN GOTO 90
BM 770 IF PEEK(764)=21 THEN GOTO 800
TD 775 IF PEEK(764)=58 THEN GOTO 790
QK 776 IF PEEK(764)=18 THEN GOTO 1600
QC 780 GOTO 740
PA 790 POKE 82,2:GRAPHICS 0:POKE 764,255:
    END
NX 800 DATA 255,255,0,6,137,6,162,0,189,2
    6,3,201,69,240,5,232,232,232,208,244,2
    32,142,105,6,189,26,3,133,205,169
UU 801 DATA 107,157,26,3,232,189,26,3,133
    ,206,169,6,157,26,3,160,0,162,16,177,2
    05,153,107,6,200,202,208,247,169,67
JZ 802 DATA 141,111,6,169,6,141,112,6,169
    ,14,141,106,6,96,172,106,6,240,9,185,1
    23,6,206,106,6,160,1,96,138,72
GO 803 DATA 174,105,6,165,205,157,26,3,23
    2,165,206,157,26,3,104,170,169,155,160
    ,1,96,0,0,0,0,0,0,0,0
ZT 804 DATA 0,0,0,0,0,76,0,0,0,34,0,68,65
    ,79,76,82,73,68,58,68,34,78,85,82,255,
    255,226,2,227,2,0,6
ZC 805 RESTORE 800:PR=0:? #6;CHR$(125);"
    WARTE EINEN MOMENT"
```

```
FG 806 IF D$(1,1)=CHR$(255) THEN 900
NO 807 FOR J=1 TO 152:POSITION 10,11:? #6
   ;152-J;" ":READ A:D$(J,J)=CHR$(A):PR=
   PR+A
ZJ 808 POSITION 3,5:? #6;"LESE DATEN EIN
   !":NEXT J
LQ 809 IF PR<>14774 THEN GRAPHICS 0:? "DA
   TENFEHLER IN 800-804":LIST 800,804:END
UF 810 GOSUB 900
BC 860 G$="ABSPEICHERN":TRAP 500:FOR J=1
   TO 2:SAVE "D:DIRLOAD":NEXT J:GOSUB 880
   :OPEN #1,8,0,"D:AUTORUN.SYS":FOR J=1 T
   O 152:PUT #1,ASC(D$(J,J)):NEXT J
OZ 861 REM ZEILE 860 MUSS ZWEIMAL (2*) SA
   VE "D:DIRLOAD" GEMACHT WERDEN.SONST FE
   HLER BEIM AUTOMATIKLADEN MOEGLICH !!!
BE 870 CLOSE #1:GOTO 600
RY 880 POKE 82,2:? :? "SCHREIBE NUN AUTOR
   UN-FILE AUF DISK -  ERMOEGLICHT AUTOMA
   TISCHES LADEN UND   STARTEN DIESES PRO
   GRAMMES."
ZW 890 RETURN
SB 900 GRAPHICS 0:POKE 710,180:POKE 16,64
   :POKE 53774,64:? :? :? :? :? :CLOSE #1
   :OPEN #1,4,0,"K:"
NI 910 ? :? "BITTE ZIELDISKETTE EINLEGEN,
   DANN  "
WY 920 ? "BITTE AUF DIE RETURN-TASTE DRUE
   CKEN :"
UU 930 TRAP ERROR:G$="DISK PRUEFEN"
WX 940 POKE 764,255
BC 950 GET #1,K:? CHR$(125):? :? :? :? :?
   :? :? :? :? :?
BX 955 GOSUB 1000:POKE 710,33
WQ 960 ? :? "SCHREIBE NUN DIR.LOADER AUF
   DIE DISK":CLOSE #1
ZT 970 RETURN
FB 1000 TRAP 1020:G$="DISK PRUEFEN"
ZJ 1010 CLOSE #1:OPEN #1,4,0,"D:DIRLOAD":
   CLOSE #1
NC 1020 TRAP 1040
GF 1030 OPEN #1,4,0,"D:AUTORUN.SYS":CLOSE
    #1:GOTO 1050
AL 1040 RETURN
QS 1050 ? "K":POKE 82,2:POKE 710,50:POKE
   709,205
UN 1060 ? :? :? :? :? :? :? :?
MU 1070 ? "DIR.LOAD-PROGRAMM BEFINDET SIC
   H SCHON AUF DIESER DISKETTE."
KK 1080 ? :?
JI 1090 ? "SOLL ALTES DIR.LOAD-FILE UEBER
   SPIELT  WERDEN ? (J/N)"
HM 1100 CLOSE #1:OPEN #1,4,0,"K:"
NO 1110 TRAP 1050
RH 1120 GET #1,K:IF K=ASC("J") THEN RETUR
   N
FZ 1130 IF K=ASC("N") THEN 600
OO 1140 GOTO 1120
TM 1160 POSITION 2,21:? "
                         "
UH 1170 POSITION 2,22:? "
                         "
EH 1180 POSITION 2,23:? "
                         ";
DH 1185 CP=0:FOR J=0 TO AZ-1:CP=CP+FL(J):
   NEXT J
TK 1190 POSITION 0,21:? "DISKETTENKAPAZIT
   AET IST :";DCP+CP:POSITION 30,21:? "KB
   YTE"
MZ 1200 POSITION 0,22:? "DAVON SIND BELEG
   T       :";CP;:POSITION 30,22:? "KBYTE
   "
OR 1210 POSITION 0,23:? "FREIER SPEICHER
   DER DISK:";DCP;:POSITION 30,23:? "KBYT
   E";:POSITION 0,21
GB 1220 POKE 764,255
HB 1230 GET #1,K
SE 1240 POSITION 0,21:? "
                         "
SZ 1250 POSITION 0,22:? "
                         "
SY 1260 POSITION 0,23:? "
                  ";:POSITION 0,21
BB 1280 RETURN
FM 1350 IF K=ASC("J") THEN K=0:RETURN
KE 1355 OPEN #1,6,0,"D:FREESECT.OR"
CP 1360 INPUT #1,E$:DCP=INT(VAL(E$)/8):CL
   OSE #1:RETURN
KO 1600 ADR=PEEK(10)+256*PEEK(11)
XW 1610 ? #6;CHR$(125):POKE 559,0
BN 1620 POSITION 0,0:? #6;"        D O S"
RT 1630 POSITION 0,1:? #6;"        -------"
WU 1640 POSITION 0,4
BT 1645 REM ADR(61987)=ZEIGER DOS AUF SEL
   FTEST (600 XL/800 XL)
UJ 1650 IF ADR=61987 THEN ? #6;"KEIN DOS
   VORHANDEN"
SL 1660 IF ADR<>61987 THEN ? #6;"DOSDISKE
   TTE EINLEGEN"
UO 1670 IF ADR<>61987 THEN ? #6;"DANN RET
   URN DRUECKEN"
ZY 1680 IF ADR=61987 THEN ? #6;"BITTE ESC
   FUER MENUE"
BB 1690 IF ADR<>61987 THEN ? #6;"ODER ESC
   FUER MENUE"
HY 1700 CLOSE #1:OPEN #1,4,0,"K:"
JV 1710 TRAP 1610:POKE 559,34
HI 1720 GET #1,K
DV 1730 IF K=155 AND ADR<>61987 THEN DOS
GS 1740 IF K=27 THEN 600
QX 1750 GOTO 1700
EU 20000 BS=PEEK(88)+256*PEEK(89)
UN 20010 LSC=BS+40*24-1
VX 20020 TRAP ERROR:CLOSE #1:OPEN #1,8,0,
   "P:":? #1;CHR$(27);CHR$(14)
CC 20030 FOR J=BS TO LSC
YD 20040 X=PEEK(J):X=X+32
OE 20050 ? #1;CHR$(X);
TJ 20055 IF PEEK(53279)=5 THEN CLOSE #1:R
   ETURN
GU 20060 NEXT J
UC 20070 CLOSE #1:OPEN #1,4,0,"K:":RETURN
```

Sicher Laden

Ein längeres BASIC-Programm ist auf Diskette gelistet worden. Beim Laden mit ENTER"D:NAME.EXT" aber stürzt der Computer ab. Dieses Programm lädt nun solche Programme meist fehlerfrei in den Speicher. Nun kann man auf das Programm wieder uneingeschränkt zugreifen. Außerdem kann man das Programm dazu benutzen, dem Computer beim Laden zuzusehen (Atari 400, 800, XL, XE und Floppy).

Arbeitsweise des Programms:
Zuerst muß in den OPEN-Befehl in Zeile 32750 der Programmname eingetragen werden. Wird das Programm dann mit RUN oder mit GOTO 32750 gestartet, werden die Programmzeilen auf dem Bildschirm ausgegeben und in den Speicher übernommen. Am Schluß wird das Programm automatisch aufgelistet. Nun sollte man die Programmzeilen 32750 bis 32756 löschen und das Programm mindestens einmal mit SAVE abspeichern. Da das Listing nur eine Variable benutzt, kann man den zusätzlichen Eintrag in die Variablentabelle vernachlässigen. Übrigens arbeitet das Programm nicht



mit Programmen, welche Zeilennummern größer als 32749 verwenden. Bei solchen Programmen gibt der Rechner einen ERROR aus. Die Zeilen bis 32749 stehen dann unversehrt im Speicher. *Holger Wasilewski*

```
KO 32750 OPEN #1,4,0,"D:FILENAME.EXT":TRA
   P 32756
BQ 32751 POKE 766,0:? CHR$(125);CHR$(29):
   POKE 766,255
JD 32752 GET #1,A:IF A<>155 THEN ? CHR$(A
   );:GOTO 32752
UV 32753 ? :? "CONT"
WL 32754 POSITION 0,0:POKE 842,13:STOP
GO 32755 POKE 842,12:GOTO 32751
XK 32756 CLOSE #1:LIST 0,32749
```

Checksummer für Atari

Da es bei der Eingabe von Atari-Listings schon öfters Probleme gab, bieten wir Ihnen einen Checksummer an, der fehlerhafte Eingaben so gut wie ausschließt. Dieser Checksummer entspricht außer zwei Verbesserungen exakt dem Checksummer, der in der amerikanischen Computerzeitschrift ANTIC jeden Monat veröffentlicht wird.

Verbesserungen:

1. Die Checksumroutine ist jetzt in Maschinensprache und somit auch schneller.
2. Wenn eine Zeile z.B. mit PRINT, END, REM oder STOP endet, stimmte die Checksum in der Zeile nicht, da beim Listing noch ein SPACE (Leerzeichen) hinzugefügt wurde und der Checksummer dieses Leerzeichen nicht beachtet. Unser Lister beachtet dies und wird kein SPACE am Ende der Zeile berechnen.
3. Der Checksummer prüft auch, ob die Reihenfolge der Buchstaben in einer Programmzeile exakt dem Listing entspricht.

Zum Beispiel PRINT „TEST" im Gegensatz zu PRINT „TSET" wird erkannt.

Tippen Sie PROGRAMM TESTER ein und speichern Sie eine Kopie auf Diskette oder Cassette ab. Geben Sie GOTO 32 000 ein und folgen Sie den Anweisungen am Bildschirm von PROGRAMM TESTER. Wenn der 2-Buchstaben-Code nicht exakt derselbe ist wie der im HC-Listing, dann ist in der soeben eingetippten Zeile ein Fehler.

Um eine schon vorher eingegebene Zeile wieder aufzurufen, tippen Sie das Sternchen (Malzeichen „*") ein, danach (ohne Leerzeichen) die Zeilennummer und dann [RETURN]. Wenn nun in der oberen Hälfte des Bildschirmes die Zeile erscheint, können Sie diese wie gewohnt editieren. Anschließend tippen Sie [RETURN].

Um das ganze Programm zu LISTEN, drükken sie [BREAK] und geben LIST ein. Zurück zu PROGRAMM TESTER mit GOTO 32 000.

Um PROGRAMM TESTER vom soeben eingetippten Programm zu entfernen, geben Sie LIST"D:FILENAME", 0,31999 [RETURN] (Cassetten-Benutzer LIST"C:",0,31999) ein. Schreiben Sie NEW, dann ENTER"D:FILENAME" [RETURN] (Cassette-ENTER"C:"). Ihr Programm ist nun im Speicher ohne PROGRAMM TESTER und kann jetzt mit SAVE oder LIST auf Diskette oder Cassette abgespeichert werden.

Benutzer, die mit BASIC XL Cartridge (der Firma O.O.S.) arbeiten, müssen SET 5,0 und SET 12,0 eingeben, bevor sie mit PROGRAMM TESTER arbeiten.

PS. Eine Routine, die den PROGRAMM TESTER von selbst löscht, ist zwar komfortabel, hat aber nicht viel Sinn, denn beim Löschen der Zeilen von PROGRAMM TESTER bleiben die Variablen-Namen, die PROGRAMM TESTER benützt, erhalten. Diese Variablen sind in einen VARIABEL-NAME-TABEL abgelegt und können nur entfernt werden, wenn wie oben beschrieben das Hauptprogramm mit LIST und ENTER von PROGRAMM TESTER getrennt wird.

```
32000 REM PROGRAMM TESTER
32005 REM
32010 DATA 34,96,215,208,207,198,203,2
30,200,230,208,208,198,6,255,238,3,208
,6,254,173,6,254,238,11,144
32015 DATA 6,253,141,6,253,109,206,165
,24,206,133,204,177,208,133,203,165,0,
160,6,255,141,6,254
32020 DATA 141,6,253,141,0,169,207,133
,104,104,204,133,104,205,133,104,104,2
03,133,1,169
32025 CLR :DIM A$(120),B$(80):CLOSE #2
:CLOSE #3
32026 RESTORE 32000
32030 H=65536:L=256:S=1789
32035 FOR X=71 TO 1 STEP -1:READ Y:B$(
X,X)=CHR$(Y):NEXT X
32040 OPEN #2,4,0,"E:":OPEN #3,5,0,"E:
"
32045 ? "↖":POSITION 11,1:? " PROGRAMM
 TESTER "
32050 TRAP 32045:POSITION 2,3:? "Gebe
Programmzeile ein"
32055 POSITION 1,4:? " ":INPUT #2;A$:I
F A$="" THEN POSITION 2,4:LIST A:GOTO
32055
32060 IF A$(1,1)="*" THEN A=VAL(A$(2,L
EN(A$))):POSITION 2,4:LIST A:GOTO 3205
5
32065 POSITION 2,10:? "CONT"
32070 A=VAL(A$):POSITION 1,3:? " ";
32075 POKE 842,13:STOP
32080 POKE 842,12
32085 ? "↖":POSITION 11,1:? " PROGRAMM
 TESTER ":POSITION 2,15:LIST A
32090 POSITION 2,16:INPUT #3;A$:IF A$=
"" THEN ? "ZEILE ";A;" WURDE GELOESCHT
":GOTO 32050
32095 C=USR(ADR(B$),ADR(A$),LEN(A$)):C
=PEEK(S)+L*PEEK(S+1)+H*PEEK(S+2)
32100 D=INT(C/676)
32105 D=C-(D*676)
32110 E=INT(D/26)
32115 F=D-(E*26)+65
32120 E=E+65
32125 POSITION 0,16:? CHR$(E);CHR$(F)
32130 POSITION 2,13:? "Wenn CODE nicht
 stimmt dann  RETURN  und editiere ob
enstehende Zeile.":GOTO 32050
```

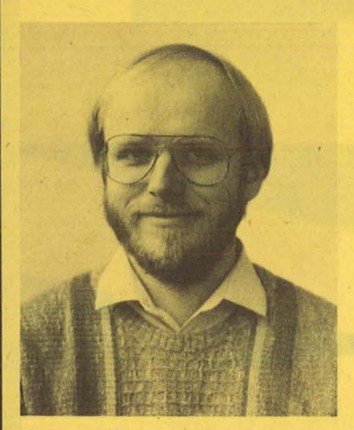
HC-EXTRA

September
1986



Pascal-News

Ansichtssache Standard oder erweitertes Pascal



Dieter Winkler, Redakteur von HC-Extra

Mit Standards ist es so ein Kreuz. Kaum hat sich ein kluger Mensch hingesetzt, ein Konzept entworfen und es auf der Basis der aktuellen technischen Möglichkeiten verwirklicht, stellt er voller Schrecken fest, daß eben diese Möglichkeiten wieder einmal entscheidend erweitert wurden.
Da hilft alles nichts, zumindest nicht in unserer computerisierten, hektischen Zeit: Das Konzept muß überdacht und mit den neuen Gegebenheiten in Einklang gebracht werden. Was sich dann daraus entwickelt, hat im Laufe der Zeit nicht mehr viel mit dem ursprünglichen Gedankenspiel zu tun.
So geht es auch Pascal. Mittlerweile ist das von Wirth 1968 konzipierte Standard-Pascal nichts weiter als ein erster Schritt in der Geschichte einer leistungsfähigen Programmiersprache; auch der rund zehn Jahre später festgehaltene ISO-Standard konnte nicht auf Dauer überzeugen. Die meisten Pascal-Dialekte ,,übererfüllen" diesen Standard bei weitem. In unserem Pascal-Teil orientieren wir uns deshalb an dieser ,,Übererfüllung", die in den meisten modernen Versionen etwa den Typ ,,String" von vornherein festlegt. In dem einen oder anderen Fall wird deshalb beim Abtippen eines Listings ein spezieller Compiler mit dem einen oder anderen Befehl nicht klar kommen; wir bemühen uns deshalb, nach und nach entsprechende Prozeduren darzulegen. Weiteres zu diesem Thema finden Sie unter ,,Pascal zum Eintippen", wo Mastermind hilft, die Übertragbarkeit von Pascal-Programmen zu verdeutlichen.

Hisoft Art

Künstlerische Betätigung am Computer mit Hilfe von Pascal erfreut sich wachsender Beliebtheit. Kein Wunder also, daß entsprechende Tools in zunehmendem Maße angeboten werden. ,,Art in Pascal" nennt sich ganz folgerichtig eine Grafik-Erweiterung für sämtliche CPC-Computer, die für Hisoft-Pascal in englischer Fassung vorliegt und von HC-Redakteur Horst Brand auf der Armstrad-Messe in London entdeckt wurde. Es kann zur Erstellung eigener Spiele und Programme in Hisoft-Pascal eingesetzt werden, was vor allem deswegen naheliegt, weil das Programm in Quelltext vorliegt. Darüber hinaus ist es ein ganz normales Grafikprogramm mit der Auswahl zwischen vier Pinselstärken, Kreis-, Ellipsen- und Rechteckerstellung und präziser Pixelansteuerung, das zur Ausnutzung der Grundgrafiken noch nicht einmal den Hisoft-Compiler voraussetzt — ratsam ist er dennoch.

Alice

Wer ,,Alice im Wunderland" kennt, dürfte die ideale Voraussetzungen besitzen, um sich mit diesem neuen Pascal-Entwicklungssystem für IBM-Kompatible auseinanderzusetzen. Konkurrenz zu Turbo oder kompatibel zu dieser mittlerweile weitverbreitetsten Pascal-Version? Wohl ein bißchen von beidem, vor allem aber ein praxisbezogenes System, das über einen Tutor sogleich in die Welt der Programmierung führt. Erfreulich ist neben der umfangreichen Dokumentation, daß 256-KByte-RAM eines Billig-PC's zur Arbeit genügen, wie Alice in der Redaktion auf einem Multitech Popular bewies. In einer der nächsten Ausgaben werden wir das als ,,Personal Pascal" angepriesene Komplettsystem unter die Lupe nehmen.

Apple II-Pascal

Wer seinen Apple liebt, gönnt ihm eine CP/M-Karte — naheliegend, daß in diesem Fall auch zu Turbo-Pascal gegriffen wird. Überarbeitet präsentiert sich jetzt Apple II-Pascal, das kein CP/M benötigt und in der neuen Version 1.3 als ,,fortgeschrittene Pascal-Entwicklungssprache" angepriesen wird. Entscheidender Vorteil im Vergleich zur Vorläufer-Version: Das 3½"-Laufwerk mit 800 KByte wird voll unterstützt, wodurch sich das Entwicklungssystem auf eine Diskette unterbringen läßt.

Pascal für Macintosh

In der letzten Folge von HC-Extra stellten wir MacAdvantage vor, die UCSD-Antwort auf den Macintosh. Aber es gibt weit mehr Neues zu berichten. Apple stellt ein rund 440 Mark teueres Entwicklungssystem vor, das in erster Linie eine Ergänzung zu Mac- und Lisa-Pascal ist und in diesen Versionen geschriebene Programme zum 68000er-Code compiliert. Damit kann auf die Macintosh-Routinen wie Operating System, Quick Draw und Printung zugegriffen werden. Dieses neue Entwicklungssystem steht in harter Konkurrenz zu TML-Pascal, das als leistungsfähiges System in Amerika weite Verbreitung gefunden hat. Es bleibt abzuwarten, welches von beiden Systemen sich in Deutschland durchsetzt.

Pascal für Enterprise

Seinem futuristischen Aussehen zum Trotz blieben die Verkaufszahlen des Enterprise weit hinter den Erwartungen zurück. Dennoch gibt es jetzt gleich zwei Pascal-Versionen für ihn: Turbo-Pascal für etwa 230 Mark und Hisoft-Pascal für knapp 70 Mark. Wie sich diese beiden Versionen unterscheiden, können Sie in der Juni-Ausgabe von HC nachlesen — beide verfügen über Editor und Compiler, der Befehlssatz von Hisoft ist gegenüber dem von Turbo eingeschränkt, doch läßt sich mit beiden Versionen gut arbeiten.

Programmierkurs

Im September veranstaltet HC einen Programmierkurs, der von Uli Kern gehalten wird, Autor der Turbo-Pascal-Reihe von Chip Special und zahlreicher Artikel zu diesem Thema. In insgesamt 16 Stunden werden zwischen 19 und 21 Uhr Einsteiger und Fortgeschrittene in Turbo-Pascal unterwiesen. Die Einsteiger sind jeweils dienstags an der Reihe, Beginn 2. 9., während Fortgeschrittenen der Donnerstag vorbehalten bleibt, Start 4. 9. 86. Anmeldungen bitte an Frau Feig unter der Anschrift: Redaktion HC, Schillerstr. 23a, 8000 München 2 oder telefonisch unter 0 89/5 14 93 43.

2 NEUE TOOLBOXEN FÜR TURBO 3.0 FANS

TURBO GAMEWORKS

Schach, Bridge und GoMoku – die drei Klassiker der Karten- und Brettspiele als (zugegebenermaßen nicht ganz vollwertiger) Ersatz für klassische Bettspiele.

Dabei ist Spielen mit diesem Paket noch das kleinste Vergnügen: Indem Sie Ihr eigenes Spiel verbessern und dem vorhandenen Schach (Bridge/GoMoku) Variationen hinzufügen, lernen Sie, wie man in Turbo-Pascal Spiele programmiert. Spaß und Unterhaltung halten sich die Waage mit dem Lerneffekt, den Sie mit diesem Programm wie nebenbei erzielen. Es ist deshalb auch vorzüglich zur Demonstration und für den Unterricht geeignet. Turbo-Gameworks ist unsere vergnüglichste Toolbox aus der Turbo-Pascal-Family und sollte deshalb jedem Anfänger mitgegeben werden. Der Profi hat Gelegenheit, sich in die Höhen der Spieltheorie zu versteigen und wer gar nicht tun will, läßt seinen Computer gegen sich selbst spielen. Das kann Ihr PC zur Erholung auch mal brauchen. Vorläufig brauchen Sie aber einen IBM PC (oder Kompatiblen) zum Spielespaß und natürlich zum Compilieren Turbo-Pascal 3.0

Für Kenner ein paar Daten, was mit Gameworks möglich ist:

- *jederzeit ein Spiel abbrechen und später wiederaufnehmen.*
- *die Spielstärke stufenlos variieren*
- *beliebige Stellungen eingeben und analysieren*
- *Zugvarianten durchspielen*
- *spezielle Modi für Mattaufgaben und Austesten neuer Strategien*
- *Auswertung eines ausführlichen Protokolls.*

***Zitat:** Georg Koltanowski, Schach-Kolumnist und Präsident der amerikanischen Schachvereinigung: »Spaß und Unterhaltung bei TURBO GAMEWORKS sind grenzenlos und vor allem kann das Programm dazu beitragen, Ihr Schachspiel unbegrenzt zu verbessern«.*

Jede Toolbox wie immer DM 225,72

TURBO EDITOR

In dieser Toolbox finden Sie wirklich alles, was Sie sich zur Umsetzung Ihrer Texte auf einem PC nur wünschen können:

MicroStar – ein unglaublich leistungsfähiges Textverarbeitungsprogramm mit allem Drum und Dran:

- *Pull-Down Menü, die komfortabelste und leichteste Art, ein Programm zu bedienen.*
- *Windows, bis zu 8 verschiedene Texte, können Sie durch 8 Bildschirmfenster gleichzeitig ansehen, verändern und montieren.*
- *Multitasking, während Sie den fertigen Text ausdrucken wollen, schreiben Sie bereits den nächsten – gleichzeitig!*
- *RAM-resident, auch in großen Dateien blättern und arbeiten Sie ohne jedes Warten.*
- *Lightning-Schnittstelle, schon heute können Sie MicroStar oder Ihr Turbo-Programm an Borlands neuester »Sensation« (Zitat aus PC Magazine) TURBO-LIGHTNING anschließen! Turbo-Lightning ist eine blitzschnelle Rechtschreibkorrektur, aber auch Sprachtraining durch Anzeige bedeutungsähnlicher Wörter. Vorläufig in englisch.*
- *und natürlich mit allen Funktionen der großen Textverarbeitungssysteme wie UNDO, Block, Zentrieren, Tabulieren, Umbruch etc. ausgestattet.*

MicroStar™ wird als funktionsfähiges Programm und im Turbo-Pascal Quellcode geliefert. MicroStar™ ist also nicht nur ein sehr edles Textsystem, sondern auch eine Riesenfundkiste für die Konstruktion und die Benutzeroberfläche Ihrer eigenen Turbo-Programme. Freie Nutzung ohne Lizenzgebühren! Vorläufig nur für IBM-PC und Kompatible, zum Compilieren brauchen Sie natürlich Turbo-Pascal 3.0!

HEIMSOETH software
Fraunhoferstr. 13 · D-8000 München 5
Telefon D-089/26 40 60/260 85 81
Telex mcm 5 212 637

Die große Welt des Turbo

Aus der umfassenden Palette von Utilities und Programmbibliotheken picken wir uns diesmal Turbo-Screen heraus.

In dieser Ausgabe von HC-Extra ist viel von Standard-Pascal die Rede. Wie dieser Standard in der Praxis aussieht, wissen zahllose Anwender bereits: Er heißt Turbo-Pascal und kostet je nach Ausführung zwischen 200 und 300 Mark. Das ist neben der unleugbaren Qualität von Turbo letztendlich ein reiner Beweis für Masse. So massiv ist mittlerweile die in die Hunderttausende gehende Verbreitung von Turbo geworden, daß es für diesen Dialekt fast mehr Zusatzprodukte als ernstzunehmende Konkurrenten gibt.

Bei den Zusatzprodukten handelt es sich in erster Linie um Software in Form von Programmsammlungen und Utilities zu jeweils speziellen Anwendungen erworben werden, etwa eins für mathematische und eins für kaufmännische Anwendungen. Die Preise der einzelnen Module orientieren sich bei diesem wie anderen Anbietern an denen von Turbo-Pascal, das heißt, sie übersteigen selten die 200-Mark-Grenze. Auch in Fragen des Quellcodes bleibt den Anbietern nichts anderes übrig, als die Karten offen auf den Tisch zu legen: Wer jetzt noch mit abgeschotteten Programmen kommt, kann – sofern es sich nicht um rein kommerzielle Software handelt – gleich einpacken. Reine Programmsammlungen haben natürlich einen entscheidenden Nachteil: Sie veralten. Einleuchtende

Außer den Sammlungen mit Programmen gibt es noch sogenannte Utility-Sammlungen. Dabei fängt die Sache natürlich an, problematisch zu werden. Gute Utilities zeichnen sich gerade durch Kürze und rechnerspezifische Anpassungen aus. Sobald aber auf die Maschinenebene zugegriffen wird, sind allgemeingültige Ansprüche einer Utility dahin. Man denke dabei etwa nur an spezielle Druckeranpassungen. Das ist der Grund, warum gerade die für den Anwender dringendsten Problemlösungen durch mehr allgemein gehaltene Sammlungen oft nicht geboten werden können.

Was nicht heißt, daß Utility-Sammlungen nicht ihr Geld wert sind. Eine der bekanntesten sind die Turbo-Power-Utilities für Diskettenverwaltung, Druckerhilfen und ähnlichem. Doch bevor wir uns in einer der nächsten Ausgaben mit diesem Gebiet auseinan-

```
Turbo Screen - Version 2.0                Copyright (C) 1985 by CID Computer-
Initialisierung Funktionstasten           Informations-Dienst GmbH, Hamburg

               Für die Initialisierung von Turbo Screen
               druecken Sie 3mal die Carriage Return-Taste.

               Bildschirmdateiname       <              >
               (X:X———X.X-X)

               Texte lesen aus Datei     <              >
               (X:X———X.X-X)

               Funktionstastenset Nr. <1>  (1/2)

Falls Sie verschiedene Tastaturen verwenden, koennen Sie zwei verschiedene Funk-
tionstastensets definieren. Es kann dann mit diesem Programm der Set gewechselt
werden, indem der aktuell zu verwendende Set angewaehlt und die Bearbeitung auf
dem naechsten Bildschirm mit 'E' beendet wird.
```

Initialisierung: *Anpassung an Tastaturen*

```
Turbo Screen - Version 2.0                Copyright (C) 1985 by CID Computer-
Initialisierung Funktionstasten           Informations-Dienst GmbH, Hamburg

Funktionstaste oder Tastenkombination und die Carriage-Return-Taste druecken

... Cursor links.......................:      ESC  K
... Cursor rechts......................:      ESC  M
... Zeichen loeschen...................:      ESC  S
... Zeichen einfuegen..................:      ESC  R
... Feldinhalt loeschen................:      ^I
... Zeichen nach links loeschen.......:      ^H
... Cursor runter/Bildschirmfreigabe..:      ESC  P
 2. Ende...............................:      ESC  O
 3. Cursor hoch / 1 Feld zurueck......:      ESC  H
 4. Sprung auf 1. Zeile / 1. Feld.....:      ESC  G
 5. Bildschirmneuausgabe...............:      ESC  I
 6. Initialisieren Bildschirmdatei....:      ESC  ;
 7. Kopie Bildschirm...................:      ESC  A
 8. Cursorposition anzeigen...........:      ESC  C
 9. Sprung auf Zeilenanfang...........:      ESC  D
10. ...................................:      ESC  Q

        B oder b: zurueck;   - : Funktionstaste loeschen   E oder e: Ende
```

Funktionstasten: *Problemlos belegt*

abgegrenzten Themenbereichen. Größtenteils liegen sie im Source-Code vor und gewähren damit Einblick in die jeweilige Programmtechnik – Anregung zum Nach- und Bessermachen.

Die vom Turbo-Pascal-Vertrieb Heimsoeth angebotenen Tools haben wir bereits in den letzten Ausgaben mehr oder weniger ausführlich besprochen; jetzt sind erst einmal die Fremdanbieter an der Reihe.

Eine der bekanntesten Programmbibliotheken ist die Turbo-Lader-Reihe. Die Idee, die ihr zugrunde liegt, ist die der Modultechnik. Ausgehend von einem Grundmodul können Module für

Programmierbeispiele und gängige Problemlösungen finden rasch weite Verbreitung in allen Medien. Entscheidend für den An-

Alles im Source-Code

wender ist aber allemal ein klares Konzept und Programme auf der Diskette, die nicht erst abgetippt zu werden brauchen. Nicht zu unterschätzen ist auch der Lerneffekt und die Möglichkeit, einzelne Programme eigenen Bedürfnissen anzupassen.

dersetzen, ist diesmal Turbo-Screen der CID-Computer aus Hamburg an der Reihe. Dabei geht es um das Problem der Datenausgabe auf den Bildschirm.

Das Sammeln von Daten und die anschließende Aufbereitung ist ein Hauptaufgabengebiet der EDV. Viele, zum Teil recht umfangreiche Programme wurden bereits in den Anfangszeiten dieses noch recht jungen Mediums geschrieben. Dabei ging es jedoch weniger um den Komfort der Bedienerführung als aus Gründen der Speicherplatzkosten um rein verwaltende Programme.

Seit einiger Zeit bietet die Hamburger Firma CID – Computer In-

-Pascals

formationsdienst – ein ganzes Paket aufeinander abgestimmter Programme zur Erfassung, Änderung und zum Löschen von Daten an. Nach einem ausführlichen Test möchte ich in zwangloser Reihenfolge die einzelnen Mitglieder dieser Turbo-Familie vorstellen.

Am Anfang steht das Sammeln, in EDV-Kreisen fachmännisch Erfassen genannt. Dabei entstehen bereits die ersten Probleme. Wer erfaßt, will und muß auch überprüfen, ob denn die Eingaben den Vorgaben in formaler Hinsicht entsprechen. Die Überprüfung numerischer Eingaben wirft Fragen auf, ob Leereingaben oder die Null erlaubt sind, ob negative Zahlen, keine oder eine bestimmte Anzahl von Nachkommastellen erlaubt sind. Bei der Überprüfung alphanumerischer Eingaben stellt die Groß- und Kleinschreibung den Datenbankprogrammierer ebenfalls vor eine Entscheidung.

und können auf beliebige MS-DOS, PC-DOS und CP/M-Systeme übertragen und dort verwendet werden.

Turbo-Screen enthält außerdem eine Reihe von Standardmoduln zum Editieren einzelner Eingabefelder sowie die komplette Ablaufsteuerung von einem Eingabefeld zum nächsten. Jedoch auch zurückgesprungen werden kann problemlos, wenn vorangegangene Eingaben geändert werden sollen. Die Module werden in die Anwendungsprogramme eingebunden.

Wichtig zur Arbeitserleichterung ist die Ausnutzung der Funktionstasten. Die Definition dieser Tasten ist maschinenunabhängig gehalten. Bei Einsatz des Systems auf

Bildschirm-Masken

einem Rechner mit einer anderen Tastatur können zwei verschiedene Funktionstastensets definiert und wechselweise installiert werden. (Pro Rechner eine Lizenz erforder-

ginnt, wird von diesem Programmteil akzeptiert. So können beispielsweise auch Funktionstasten mit Ansi-Codes oder solche mit 8-bit-Codes verwendet werden.

Die eigentliche Maskenerstellung ist mit wenigen Tastendrukken erledigt. Man bekommt einen leeren Bildschirm angeboten und beginnt mit dem Plazieren der Namen der Ein- und Ausgabefelder. Auf Tastendruck übernimmt das Programm die Anfangspunkte und speichert sie in einer Datei ab.

Mit dem zweiten Programm dieser Familie, Turbo-Screen-Plus, kann man sogar die Position der Ein- und Ausgabefelder in diese Datei übernehmen. Nach und nach wächst menügesteuert die komplette Maske der gewünschten Datenbank im Dialog auf dem Bildschirm. Jede Phase kann dokumentiert werden.

Turbo-Screen-Plus bietet zusätzlich per Menüführung die Generierung eines Bildschirmmoduls, einer Bildschirmdatei und eines Ausdruckes der Bildschirmbeschreibung.

Hier liegt meiner Meinung nach auch ein Schwachpunkt des sonst



Hauptmenü: *Reichhaltige Auswahl*

Turbo Screen+ Copyright (C) 1985 by CID-Computer-
Hilfe: Hauptauswahl Informations-Dienst GmbH, Hamburg

Feld 1: Auswahl

Waehlen Sie bitte einen der aufgefuehrten Punkte aus und geben Sie die zugehoerige Nummer in diesem Eingabefeld ein.

Neben den Auswahlpunkten 1-3 und 7, die Programme von Turbo Screen+ aufrufen, finden Sie im Menue auch Punkte, die Programme des Grundsystems Turbo Screen ansprechen. Wenn Sie moechten, dass nach Ausfuehrung dieser Programme wieder in das Menue von Turbo Screen+ verzweigt wird, beachten Sie bitte die Anweisungen im Handbuch, die sich auf die Aenderung der Turbo Screen-Programme beziehen.

Feld 2: Aufrufen von Laufwerk

In diesem Feld kann das Laufwerk ausgewaehlt werden, auf dem sich das zu startende Programm befindet. Ist es das voreingestellte Laufwerk, so ist eine Eingabe nicht erforderlich. In diesem Fall koennen Sie auch die Auswahl auf Feld 1 mit der Bildschirmfreigabetaste (Cursor nach unten) abschicken. Dann wird Feld 2 erst garnicht angesprungen.

—> Taste druecken.

Hilfe: *Zum schnellen Einstieg*

Hier mußten bisher immer umfangreiche Prüfroutinen programmiert werden. Turbo-Screen nimmt diese Arbeit ab. Umfassend sind die kleinen und größeren Nöte eines „eigentlich Nur-Anwenders" bedacht worden und in eine standardisierte Dialogeingabesteuerung für Turbo-Pascal-Programme umgesetzt worden. Das Programm ist ein Maskeneditor, mit dem Bildschirmmasken am Computer entworfen, gespeichert und auch modifiziert werden können. Diese Bildschirmsteuerung ist leicht in andere Sprachen wie BASIC, COBOL oder PL/I übertragbar.

Die einmal gespeicherten Masken werden Programmbestandteil

lich). Neben dem Bildschirmeditor und den für die Eingabesteuerung notwendigen Standardmodulen werden Konvertierungsprogramme zur Umsetzung von Bildschirmdateien in Textdateien und umgekehrt mitgeliefert. So können die selbsterstellten Masken zwischen verschiedenen Rechnern mit unterschiedlichen Betriebssystemen übertragen werden. (Lizenz!)

Die Arbeit mit Turbo-Screen gestaltet sich einfach und übersichtlich. Als erstes Menü erscheint die Anpassung an die Tastatur. Dabei fällt auf, daß die Anpassung praktisch auf jeder Tastatur möglich ist. Jede Zeichenfolge, die mit einem nichtdarstellbaren Zeichen be-

sehr gut durchdachten Programms. Wer wie ich nur einen 80-Zeichen-Drucker benutzt, muß seinen Drucker vorher auf komprimierte Schrift umschalten. Dazu muß man das Programm verlassen oder von vornherein bei der Arbeit mit Turbo-Screen diese Umschaltung vornehmen. Ich wünsche mir eine kleine Anpassungsroutine, in der ich die Steuersequenz meines Druckers dem Programm mitteilen kann, so daß die Umschaltung auf „Condensed Mode" sowie auch die Rücksetzung vom Programm im Bedarfsfall ausgeführt wird. Ansonsten ist den Programmen Professionalität zu bescheinigen.

Wolfgang Börner/dw

Pascal leicht gemacht

In der Fernuni kann sich jeder einschreiben, der Pascal lernen will – Abschlußprüfung eingeschlossen

Es gibt viele Möglichkeiten, Pascal zu erlernen. Man kann Bücher lesen, Zeitschriftenserien verfolgen oder Lernprogramme bearbeiten. Diese Möglichkeiten haben gravierende Nachteile gemein: einerseits fehlt eine verläßliche Kontrolle des erworbenen Wissens, andererseits kann man seine Kenntnisse (beispielsweise bei Bewerbungen oder beim Studium) nicht ohne weiteres nachweisen.

Die reinen Hobbyprogrammierer haben es auf den ersten Blick da noch relativ leicht: Nach dem Erwerb von Compiler und Editor wird einfach drauflosgehackt, wobei Lücken anhand der mittlerweile zahlreich angebotenen Literatur Stück für Stück geschlossen werden. Aber gerade das wird oft zum Problem. Verläßliche Hinweise auf den Wert unterstützender und zumeist teurer Literatur erhält man erst, wenn das gute Stück zumindest halb durchgearbeitet ist.

Nicht selten sinkt der Wert eines Buches dadurch, daß es sich auf veraltete Versionen und Sprach-Dialekte bezieht – etwa, wenn die Turbo-Version 3.0 außer acht gelassen wird und dafür breit Nachteile der Version 2.0 besprochen werden. Eine Möglichkeit zur Abhilfe besteht darin, zusätzlich zum Buch eine Programm-Diskette anzubieten. Von dieser Grundidee gehen beispielsweise der Turbo-Tutor der Firma Heimsoeth und die preiswerte Turbo-Pascal-Reihe von CHIP-Special aus. Typisch für solche Werke ist, daß der Quellcode ausgedruckt im Begleittext steht.

Das ist auch notwendig, wenn Pascal-Dialekte nicht rechnerspezifisch sind und auch Besitzer sogenannter Exoten-Rechner auf ihre Kosten kommen sollen. Naturgemäß werden begleitende Disketten immer nur für ein begrenztes Rechner-Spektrum angeboten. Aber es geht natürlich auch umgedreht.

So bietet der Verlag Markt&Technik ein neues Werk mit dem Titel ,,Pascal mit dem C64" an. Im Mittelpunkt steht der Rechner und ein speziell auf ihn zugeschnittener Pascal-Compiler und Editor; letztere sind im Kaufpreis von 52 Mark enthalten. Der Compiler entspricht dem Standard und ist damit etwas für Pascal-Puritisten; für weitergehende Arbeit ist er kaum geeignet. Leider ist seine Bedienungsanleitung etwas dürftig und schmälert den Nutzen als Einsteigersystem.

Pascal im Fernstudium – ein Erlebnisbericht

Eine interessante Alternative bietet das Fernstudium, in Deutschland zum Beispiel an der Fernuniversität in Hagen. Neben zahlreichen anderen Kursen der verschiedensten Fachgebiete wird dort – vom Fachbereich Mathematik/Informatik – auch ein Pascal-Kurs angeboten. Voraussetzungen zum Studium existieren nicht: formell heißt man allerdings ,,Gasthörer", solange man keine ,,Hochschulzugangsberechtigung", sprich: Abitur, hat. Über eine Anerkennung der erbrachten Leistungen entscheidet im Studiumsfall die jeweilige Fachschaft.

Nachdem ich bereits ein Semester lang den Kurs ,,Programmierung" belegt hatte, schrieb ich mich im Sommersemester 1983 zum Kurs ,,Pascal" ein. Bald darauf erhielt ich die erste Kurseinheit des aus ingesamt sieben Teilen bestehenden Kurses.

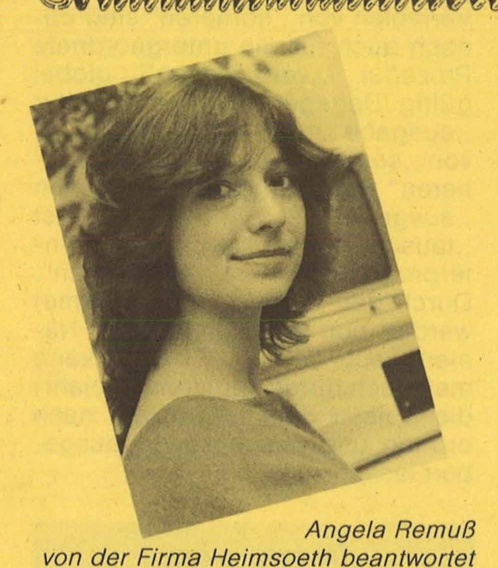
Jede Einheit besteht aus einem etwa 50seitigen Textteil, der thematisch abgeschlossen und in sich übersichtlich gegliedert ist. Darin integriert sind Selbsttestaufgaben, die zum frühzeitigen Erkennen von Unsicherheiten wichtig sind. Dazu kommen Übungsaufgaben, die einerseits Schwerpunkte der jeweiligen Einheit aufzeigen und andererseits die Überprüfung des Erlernten erleichtern. Schließlich gehören zu jeder Einheit Testaufgaben, die selbständig, wenn möglich unter Klausurbedingungen, gelöst werden und anschließend nach Hagen geschickt werden sollen. Die Lösungen werden dort EDV-mäßig erfaßt und bewertet, das Ergebnis dem Studenten mitgeteilt. Ein bestimmter Prozentsatz richtiger Lösungen ist Voraussetzung zur Teilnahme an der Abschlußklausur. Alle sieben Einheiten zusammen ergeben – nicht zuletzt dank des mitgelieferten Glossars – ein recht brauchbares Werk über die Sprache Pascal.

Da ich bereits einige Grundkenntnisse der Sprache Pascal besaß, glaubte ich mich über den Hinweis, die Einheiten müßten durchgearbeitet, nicht nur gelesen werden, hinwegsetzen zu können und hatte infolgedessen binnen zweieinhalb Stunden die Kurseinheit ganz durchgelesen, aber nur die Hälfte behalten. Dementsprechend sah auch die Beurteilung meiner Einsendeaufgaben aus, die ich einige Wochen später in Händen hielt.

Die nächsten Einheiten bearbeitete ich dann schon intensiver, wodurch ich etwa vier bis fünf Stunden je Einheit brauchte; das reichte für etwa 80 Prozent bei den Einsendeaufgaben. Damit lag mein Aufwand allerdings immer noch unter dem vorgesehenen von vier Stunden pro Woche oder etwa 15 Stunden pro Kurseinheit. Bemerkenswert ist, daß gewisse mathematische Kenntnisse (etwa der Vektorbegriff) vorausgesetzt werden, die ich zusätzlich erwerben mußte. Die im Verlauf des Kurses beispielhaft gelösten Probleme sind meist mathematisch-naturwissenschaftlicher Art, so daß ich gleichzeitig auf diesem Gebiet einige mir neue Sachverhalte erfuhr. Ende des Jahres fand schließlich in Köln die Abschlußklausur statt, die ich mit ,,gut" bestand.

Zusammenfassend bleibt festzustellen, daß man für 75 Mark nicht nur ein Lern- und Nachschlagewerk über die Sprache Pascal erhält, sondern darüber hinaus Einblick in das Studienfach Informatik gewinnt. Sicherlich ist ein solches Fernstudium wesentlich effektiver als das Lesen eines Buches, doch ist man im zeitlichen Zugzwang und muß erheblich mehr Zeit investieren. Insgesamt scheint mir das Fernstudium jedenfalls für den ernsthaften Interessenten eine vorteilhafte Alternative, zumindest aber eine sinnvolle Ergänzung zu sein. *Jörg Tegeder/dw*





Angela Remuß
von der Firma Heimsoeth beantwortet
exklusiv Fragen unserer Leser.

Aus dem Stoß der vorliegenden Fragen haben wir diesmal einige herausgegriffen, die typische Problemstellungen wiedergeben. Dabei kommt auch ein Druckfehler aus HC 5/86 zur Sprache, der ausgerechnet beim Patchen zur Verzweiflung führen kann. Anstelle eines FC ist an der Speicherstelle 2E8C ,,1B" einzugeben – wer hätte das gedacht. Und noch eine erfreuliche Mitteilung: Für den DIV/MOD-Fehler unter Turbo-Pascal Version 3.0 für CP/M 80 ist ein Patch gefunden worden, den wir in der nächsten Ausgabe vorstellen.

CPC 664 mit Speichererweiterung von Data Media steigt beim Compilieren aus. Wie ist dem abzuhelfen?
Frage von Walter Ganter aus Düren

Nach Aufruf der Speichererweiterung durch SYS und RUN ,,TURBO" passiert folgendes: Tritt beim Compilieren ein Fehler auf, der sich durch Rücksprung ins Turbo-Programm durch Drücken der ESC-Taste beheben lassen würde, so blockiert der Schneider CPC 664. Nur durch einen Kaltstart läßt sich das System wieder starten. Die in HC 5, Seite 86, vorgeschlagene Lösung funktioniert nicht, da ohne Speichererweiterung die ESC-Taste eingesetzt werden kann.
Die Lösung ist dennoch denkbar einfach: In HC 5/86 hat sich ein Fehler eingeschlichen. Es muß nicht in der Speicherzelle 1B der Wert in FC geändert werden, sondern in der Speicherzelle 2E8C muß der Wert in FC 1B geändert werden (beim DDT üblichen Offset von 100 hexadezimal).

Fragen an unsere Hotline

Falls Sie irgendwelche Probleme mit Turbo-Pascal oder bestimmte Fragen zur Anwendung dieser Sprache haben, können Sie sich direkt an unsere Hotline wenden. Schreiben Sie an:

Vogel-Verlag
Redaktion HC
Stichwort: Pascal-Hotline
Schillerstr. 23a
8000 München 2

Warum wird beim Schneider CPC 6128 der freie Diskettenplatz falsch angegeben?
Frage von Berthold Bollinger aus Hürth und Hans Maas aus Bremen

Wählt man beim CPC 6128 unter CP/M 3.0 im Hauptmenü die Funktion ,,D" für ,,Directory anzeigen", so erscheint unter dem Disketteninhaltsverzeichnis nach der Mitteilung ,,Bytes remaining on A:" eine Zahl, die eindeutig nicht mit dem freien Speicherplatz übereinstimmt, sondern um etwa 64 KB darüber liegt. Das liegt daran, daß Turbo-Pascal an CP/M 2.2 angepaßt wurde und die Version 3.0 nicht ganz aufwärtskompatibel zur Version 2.2 ist. Der Fehler bei Berechnung des freien Speicherplatzes wirkt sich jedoch nicht auf das Speichervermögen aus. Nur die Anzeige ist fehlerhaft. Dafür wurde bislang leider noch keine Korrektur gefunden.

Kann Turbo-Pascal auch mit einer RAM-Disk zusammenarbeiten?
Frage von K. Leimer aus Düsseldorf

Prinzipiell ist eine RAM-Disk nichts anderes als ein Teil des Hauptspeichers, der wie ein Disketten-Laufwerk angesprochen wird. Es bietet sich daher sogar an, Turbo-Pascal nach Möglichkeit mit einer RAM-Disk zusammenarbeiten zu lassen. Da eine RAM-Disk keine mechanischen Teile besitzt, ist das Schieben von Daten von und nach Disk viel schneller möglich als mit einem herkömmlichen Laufwerk. Entscheidend bemerkbar macht sich dieser Vorteil, wenn viel mit Overlays und anderen nachladbaren Programmteilen gearbeitet wird. Kleine Overlays stehen praktisch sofort zur Verfügung. Eins sollte jedoch bei der Arbeit mit RAM-Disks beachtet werden: Sie ist kein Permanent-Speicher, das heißt, alle Daten gehen beim Abschalten des Computers verloren, sofern sie nicht vorher auf eine Diskette oder Festplatte abgespeichert wurden. Besonders elegant: alle wichtigen Programmteile beim Systemstart automatisch – zum stetigen Zugriff – auf die RAM-Disk zu schieben.

Wie werden in Turbo-Pascal die BASIC-Befehle INP und OUT geschrieben?
Frage von Raimund Pourvoyeuer aus Reutte

Unter Turbo-Pascal gibt es ein vordefiniertes PORT-ARRAY, mit dem sich der Datenport des 8086/88 ansprechen läßt. Über dieses Array können die Funktionen INP und OUT ersetzt werden.

Pascal-Lernen mit HC

Teil 6: Der richtige Einsatz von Prozeduren

In der letzten Folge unseres Pascal-Kurses haben wir den Umgang mit Prozeduren kennengelernt. Darunter versteht man eine beliebige Folge von Anweisungen, die mit einem Namen aufgerufen werden kann; das Prinzip ist in dem ,,Beipackzettel" nochmals dargestellt.

Eine Reihe von Prozeduren sind bereits im Sprachumfang von Pascal enthalten. Beispielsweise verbirgt sich hinter ,,writeln" ein derartiges Unterprogramm. Im Gegensatz zu den vom Benutzer definierten Prozeduren sind bei dieser Druckprozedur jedoch Art und Anzahl der dem Unterprogramm übergebenen Werte (Parameter) nicht festgelegt.

Zur Erinnerung: Am Beispiel einer Liste der Daten von Bundesligaspielern haben wir festgestellt, daß Programme durch Verwendung von Prozeduren übersichtlicher geschrieben werden können.

Schachteln von Prozeduren

Eine Prozedur kann auch eine andere Prozedur aufrufen, wie das Beispiel ,,sortieren" zeigt, in dem die Spieler nach einem bestimmten Merkmal sortiert werden können. Das Sortieren durch fortlaufendes Vertauschen haben wir bereits näher erläutert. Das Sortierprogramm wird sehr viel übersichtlicher, wenn man eine Prozedur ,,vertauschen" einführt, die im Bedarfsfall die gesamten Daten zweier Spieler austauschen kann.

Im Vereinbarungsteil von ,,sortieren" werden ,,index" und ,,durchlauf" als ganze Zahlen festgelegt. Danach wird die Prozedur ,,vertauschen" erklärt. In der dazugehörenden Zeichnung wird die Schachtelung des gesamten Programms veranschaulicht. Alle Variablen von ,,sortieren" sind danach auch für die untergeordnete Prozedur ,,vertauschen" global gültig. Dagegen kann die Prozedur ,,ausgabe" nicht auf die Variablen von ,,sortieren" zugreifen, da ,,sortieren" kein Unterprogramm von ,,ausgabe" ist. Andererseits ist ,,tausch1" nur lokal gültig im Unterprogramm ,,vertauschen". Durch den Aufruf sortieren(name) werden nun alle Spieler dem Namen nach sortiert. Ebenso kann man auch durch sortieren(gebjahr) die Spieler dem Lebensjahr nach ordnen und anschließend ausgeben lassen.

Gebrauchsinformation!
Sorgfältig lesen!

PROZEDUR®

Eigenschaften:	Ein seit vielen Jahren bewährtes Mittel gegen unstrukturiertes Programmieren.
Zusammensetzung:	Mit Werte- oder Namensparametern in Klammern.
Anwendungsgebiete:	Lange oder unübersichtliche Programme.
Nebenwirkungen:	Vorsicht bei gleichzeitiger Verwendung von globalen Variablen!
Besondere Hinweise:	Gelegentlich können unklare Fehlersymptome durch Verwechslung von Werte- und Namensparametern auftreten.

```
PROCEDURE sortieren(wonach:merkmal);
(* Zum Einbau in das Programm 'bundesliga'*)
 VAR index,durchlauf:integer;

PROCEDURE vertauschen(a,b:integer);
 (* Vertauscht die Daten der Spieler mit den
    Nummern a und b *)

 VAR  merkm : merkmal;
    tausch1 : zkette;

BEGIN
 FOR merkm:=name TO gebjahr DO
  BEGIN
   tausch1:=spieler[a,merkm];
   spieler[a,merkm]:=spieler[b,merkm];
   spieler[b,merkm]:=tausch1;
  END;
END;  (* von vertauschen *)

BEGIN (* von sortieren *)
 FOR durchlauf:=1 TO spielerzahl DO
  BEGIN
   FOR index:=1 TO spielerzahl-1 DO
    BEGIN
     IF spieler[index,wonach]>spieler[index+1,wonach] THEN
      vertauschen(index,index+1);
    END;
  END;
END;
```

Wert- oder Namensparameter?

In allen bisherigen Programmen haben wir jeweils feste Werte an ein Unterprogramm übergeben. Dies kann zu auf den ersten Blick merkwürdigen Effekten führen, wie das ,,parameterdemo"-Programm zeigt.

```
PROGRAM parameterdemo;
 (* Fehlerhaftes Programm *)
 VAR zahl:integer;

PROCEDURE quadrat(a:integer);
 BEGIN
  a:=a*a;
 END;

BEGIN
 write('Zahl eintippen!  ');
 readln(zahl);
 quadrat(zahl);
 writeln('quadriert=',zahl:3);
END.
```

In der Prozedur ,,quadrat" wird offenbar die ganze Zahl a mit sich selbst multipliziert und das Ergebnis wieder a zugewiesen. Im Hauptprogramm müßte zu einem eingetippten Wert ,,zahl" eigentlich dessen Quadrat ausgedruckt werden. Der Beispielausdruck ergibt aber:

Zahl eintippen! 6
quadriert= 6

Wo liegt der logische Fehler? Die Prozedur faßt a als einen festen Wert und nicht als eine Variable auf. Daher kann durch a:=a*a

die ganze Zahl a auch nicht verändert werden. Da wir dies aber erreichen wollen, müssen wir das Unterprogramm anweisen, a als Variable aufzufassen. Mit der abgeänderten Vereinbarung procedure quadrat(var a:integer) ergibt sich tatsächlich das richtige Ergebnis. Probieren Sie es aus!

Es gibt also zwei verschiedene Arten der Parameterübergabe. Entweder wird nur ein Wert übergeben (Werteparameter), der von der Prozedur nicht geändert werden kann, oder der Parameter wird durch den Zusatz ,,var" als Variable erklärt. Im letzten Fall wird dem Unterprogramm gewissermaßen der Name des Parameters durchgereicht (Namensparameter). Dadurch kann die Prozedur den Wert verändert an das rufende Programm zurückgeben.

Doch nun zurück zu der Aufgabe, die im letzten Teil gestellt wurden. Als Übungsaufgabe sollte eine Prozedur ,,loeschen" erstellt werden. Wir wollen damit den Spieler mit der Nummer a aus der Liste entfernen. Dazu vertauschen wir einfach die Daten des Spielers mit der höchsten Nummer mit denen des zu löschenden Spielers. Für diesen Zweck bietet sich die bereits vorhandene Prozedur ,,vertauschen" an. Zum Schluß erniedrigen wir die Anzahl der Spieler.

```
PROCEDURE loeschen (a:integer);

 BEGIN
  vertauschen(a,spielerzahl);
  spielerzahl:=spielerzahl-1;
 END;
```

Dennoch scheitert der Versuch bereits beim Übersetzen: Der Compiler bemängelt, daß die Prozedur ,,vertauschen" in ,,loeschen" nicht erklärt ist. In der Tat haben wir ,,vertauschen" nur in den Erklärungsteil von ,,sortieren" eingebaut, weil dies für den ursprünglichen Verwendungszweck auch nur dort erforderlich war. Wenn man stattdessen ,,vertauschen" im Erklärungsteil des Hauptprogrammes einbaut, funktioniert das Löschen eines Spielers einwandfrei. Man muß also nicht nur die Datenstrukturen, sondern auch die Schachtelung der Prozeduren sorgfältig planen.

Prozeduren mit Rückgabegarantie

Bei einer Reihe von Programmen unseres Kurses ist uns schon ein Ausdruck wie ,,length(wort)" begegnet, mit dem man die Anzahl der Buchstaben in der Zeichenkette ,,wort" bestimmen kann. Offenbar kann es sich dabei trotz der ähnlichen Schreibweise nicht um eine Prozedur handeln, da der Ausdruck keine Anweisung bedeutet, sondern wie eine reelle Zahl behandelt wird. Hier haben wir es mit einer sogenannten ,,Funktion" zu tun, durch die einer Variablen ein Wert des einfachen Datentypes zugeordnet wird:
Variable: wort → length(wort)
Typ: zeichenkette → integer length (wort) steht also für eine ganze Zahl. Damit sind auch Verknüpfungen wie 5 + length(wort) möglich. Würde es sich dagegen um eine Prozedur handeln, müßte sich stets ein Semikolon hinter dem Aufruf befinden. Im Verlauf unseres Kurses haben wir schon mehrfach von derartiger eingebauter Funktion Gebrauch gemacht. Vielleicht erinnern Sie sich noch:
Sofern die Funktion ,,random" in der entsprechenden Version implementiert ist, steht random(5) beispielsweise für eine Zufallszahl zwischen 0 und 5. Hinter random(zahl) steckt die Zuordnung
Variable: zahl → random(zahl)
Typ: integer → integer
Ein weiteres Beispiel ist die im Sprachumfang vieler Pascal-Versionen enthaltene odd-Funktion, die angibt, ob eine Integerzahl gerade ist. So steht odd(4) für ,,true", während odd(27) ,,false" ergibt. Durch die Funktion mit dem Namen ,,odd" wird also einer Zahl vom Typ Integer ein logischer Wert zugeordnet:
Variable: zahl → odd(zahl)
Typ: integer → boolean

```
PROGRAM wertetabelle;
 VAR x,xanfang,xende:real;
     xschritt       :real;
 (*-------------------------------------*)
 FUNCTION f(s:real):real; (* Festlegung  *)
  BEGIN                   (* der Funktion*)
   f:=4*s*s-4             (* namens f    *)
  END;                    (* Ende d. Def.*)
 (*-------------------------------------*)
BEGIN        (* Beginn des Hauptprogrammes *)
 write('Anfangswert  ');readln(xanfang);
 write('Schrittweite ');readln(xschritt);
 write('Endwert      ');readln(xende );
 writeln;
 writeln('   Wertetabelle ');
 writeln ('    x          f(x) ');
 writeln ('-------------------');
 x:=xanfang;              (* Erster x-Wert*)
 WHILE x<= xende DO
 BEGIN
  writeln(x:7:3,'     ',f(x):7:3);
  x:=x+xschritt;          (* Neuer x-Wert *)
 END;
END.
```

Schnell noch ein nicht ganz ernst zu nehmendes Beispiel:
Variable: Leser → HC
Typ: interessiert → gut
Aber wie bei den Prozeduren wollen wir uns auch selbst Funktionen definieren.

Mit diesem kleinen Programm kann man sich für die Funktion f(s):=4∗s∗s−4 eine Wertetabelle ausdrucken lassen. Neu hieran ist die im Listing eingerahmte Definition der Funktion mit dem Namen f, die wie bei Prozeduren auch im Erklärungsteil stehen muß. Diese Kopfzeile besagt, daß f(s) eine Zahl vom Typ ,,real" ist. Beim Aufruf muß in der Funktionsklammer ebenfalls eine Kommazahl stehen. Die Vereinbarung für f wird in der Druckanweisung am Ende des Programmes benutzt. Damit ergibt sich nach der Eingabe von Anfangswert, Schrittweite und Endwert beispielsweise folgendes Druckbild:

Wertetabelle

x	f(x)
3.000 3.200 3.400 3.600	32.000 36.960 42.240 47.840 usw.

Auswerten von Meßreihen

Einer Funktion können auch mehrere Parameter übergeben werden, wie das folgende Beispiel ,,funktionsdemo1" zeigt, bei dem zu eingetippten Meßwerten deren Mittelwert ausgedruckt wird. Die Kopfzeile der Funktion besagt, daß ,,mittelwert" für eine Kommazahl

```
PROGRAM funktionsdemo1;
 TYPE messzahl = 1..20;
      messwert = ARRAY[messzahl] OF real;
 VAR          messung : messwert;
    messnr,messungszahl : messzahl;
                miwert : real;
(*----------------------------------------------------- *)
FUNCTION mittelwert (mess:messwert;n:messzahl):real;
 VAR nummer:messzahl;
     summe :real;
 BEGIN
  summe:=0;
  FOR nummer:=1 TO n DO           (* Messwerte werden      *)
   summe:=summe+mess[nummer];  (* addiert.Die Summe      *)
  mittelwert:=summe/n;         (* wird durch die Zahl    *)
 END;                          (* der Messungen geteilt*)
(*------------------------------------------------------*)
BEGIN                          (* Beginn des Hauptprogrammes *)
  write ('Zahl der Messwerte:= ');
  readln (messungszahl);
  FOR messnr:=1 TO messungszahl DO
   BEGIN
    write (messnr:3,'. Wert:= ');
    readln (messung[messnr]);
   END;                        (* Messwerte sind  eingelesen *)
  miwert:= mittelwert(messung,messungszahl);
  writeln ('Mittelwert:= ', miwert:7:3);

  (* Nun noch eine mathematische Merkwuerdigkeit *)
  FOR messnr:=1 TO messungszahl DO
  messung[messnr]:=messung[messnr]-miwert;
  write ('Neuer Mittelwert:= ');
  writeln (mittelwert(messung,messungszahl):7:3);
 END.
```



steht. Der durchgereichte Parameter „mess" ist ein Array von n reellen Zahlen. Im Hauptprogramm werden zunächst die Meßwerte eingelesen. Im Anschluß daran erfolgt der Aufruf der Funktion „mittelwert", deren Wert noch in einer besonderen Variablen „miwert" abgespeichert wird.

Am Ende des Programms werden die Meßwerte dadurch geändert, daß jeweils die Differenz des Meßwertes zum Mittelwert genommen wird. Hätten wir nicht zuvor die Variable „miwert" eingeführt, müßte man für jede Differenz die Funktion „mittelwert" aufrufen, was natürlich die Laufzeit des Programmes verlängert hätte. Das Ergebnis dieser zweiten Berechnung ist merkwürdigerweise für alle Meßreihen 0. Können Sie sich dies mathematisch erklären?

Für die Funktionsschreibweise gibt es in Pascal allerdings einige wenige Ausnahmen, wie die Funktion „mod", die zu zwei Integerzahlen a und b den ganzzahligen Rest beim Teilen von a durch b angibt. So ergibt sich für a = 50 und b = 6 als Rest beim Teilen 2, denn 50=6∗8+2. Nach der bisherigen Grammatik müßte man dies eigentlich als mod(50,6) bezeichnen. In Anlehnung an die mathematisch übliche Schreibweise benutzt man jedoch in Pascal 50 mod 6. Überlegen Sie sich bitte, daß 60 mod 7 = 4 ist!

Prima Zahlen – Primzahlen

Mit der mod-Funktion kann man schnell prüfen, ob eine Zahl durch eine andere teilbar ist. Dies ist dann der Fall, wenn der Rest bei der Division Null ist. Da 16 durch 2 teilbar ist, gilt 16 mod 2 = 0.

Eine weitere Ausnahme ist die Funktion „div", mit der man das ganzzahlige Ergebnis bei Dividieren zweier Integerzahlen bestimmt. So ist 26 div 9 = 2 und 86 div 12 = 7. Die Funktionen mod und div dürfen nur mit Parametern benutzt werden, deren Werte zwischen 0 und 32767 liegen. Mit der Funktion mod läßt sich prüfen, ob eine Zahl eine Primzahl ist. Zur Erinnerung: 37 ist beispielsweise eine Primzahl, da sie nur durch sich selbst und 1 teilbar ist. Verabredungsgemäß ist übrigens 1 keine Primzahl.

Die Funktion „primzahl" ordnet jeder Zahl x den Wahrheitswert „false" oder „true" zu, je nach-

```
PROGRAM primzahlsuch;
 TYPE ganzzahl=1..32767;
 VAR zahl,anfang,ende :ganzzahl;

 FUNCTION primzahl(x:ganzzahl):boolean;
  (* Ist x eine Primzahl ? *)
  VAR teiler:integer;
      gefunden:boolean;
  BEGIN
   gefunden:=false;
   FOR teiler:=2 TO x-1 DO
    IF (x MOD teiler)=0 THEN gefunden:=true;
   IF (gefunden=true ) OR (x=1)  THEN
    primzahl:=false
   ELSE
    primzahl:=true;
  END;

BEGIN
 write ('Primzahlsuche von ');read (anfang);
 write ('  bis  ');readln(ende);
 writeln ('Folgende Zahlen sind Primzahlen:');
 FOR zahl:=anfang TO  ende DO
 BEGIN
  IF primzahl(zahl) THEN
   write (zahl:8);
  END;
END.
```



dem, ob x prim ist oder nicht. Dazu werden alle denkbaren Teiler von 2 bis x−1 durchprobiert. Wenn unter diesen bei keinem der Divisionsrest 0 ist, handelt es sich um eine Primzahl. Das Programm läuft natürlich schneller, wenn wir nur die Teiler von x bis (x div 2) prüfen, denn beispielsweise bei 37 kann es keinen größeren Teiler als 18 geben. Noch ein Tip: Extrem schnell läuft das Programm, wenn man nur die Teiler von 2 bis Wurzel aus x prüft. Auf die mathematische Begründung des Verfahrens wollen wir hier aber verzichten. Dazu muß man nur schreiben:
for teiler:=2 to trunc(sqrt(x)) do
Dabei wird mit sqrt die Wurzel berechnet, deren Nachkommastellen mit trunc abgeschnitten werden. Dies ist wegen der erforderlichen Typgleichheit der Variablen in einer for-Schleife nötig.

Lange Integerzahlen

Im Programm longint (Abkürzung von: long integers) wird abschließend noch einmal das Zusammenspiel von Prozeduren und Funktionen gezeigt. Dabei werden Integerzahlen mit bis zu 40 Stellen als Zeichenketten abgespeichert. Das zugehörige Hauptprogramm ist besonders übersichtlich und auf Anhieb lesbar. Man kann nicht nur Variablen, sondern auch Konstanten vereinbaren. Dies geschieht durch „const". Dabei muß die Typangabe entfallen, denn der Pascal-Compiler erkennt beispielsweise aus der Vereinbarung „const pi=3.1414;", daß mit pi eine Kommazahl gemeint ist. Der Wert einer Konstanten kann im laufenden Programm nicht geändert werden. Mit der Funktion val wird ein Stringausdruck in eine Zahl umgewandelt. Die Syntax ist in Turbopascal val (string, val, code), wobei die boolesche Variable „code" angibt, ob eine Umwandlung möglich ist. „A34" läßt sich nicht umwandeln!

Die ebenfalls neu auftretende Funktion „copy" ordnet einer Zeichenkette einen Teil einer (anderen) Zeichenkette zu. So ist beispielsweise copy(e,4,5) eine Zeichenkette, die aus 5 Buchstaben der Stringvariablen e, beginnend ab dem vierten Zeichen von e besteht. In der Funktion „plus" werden die Zahlen ziffernweise unter Berücksichtigung des Übertrages addiert. Als Aufgabe ist diesmal das Hauptprogramm von „primzahlsuch" so umzuschreiben, daß nur dann eine Primzahl ausgedruckt wird, wenn die um zwei größere Zahl ebenfalls prim ist. Solche Zahlen wie 17 und 19 nennt man Primzahlzwillinge. *Thomas Geise*

```
PROGRAM longint;
{ Verarbeitet Integervariablen beliebiger Laenge
  (Voreinstellung durch 'null' und
  'laenge' : 40 Stellen .Empfehlung:laenge<80 ).
  Die Zahlen werden ziffernweise als Zeichenketten
  bearbeitet und in der Zeichenkette stets
  rechtsbuendig abgespeichert.                  }

 CONST null=
   '0000000000000000000000000000000000000000';
 laenge=40;
 TYPE zahl=STRING[laenge];
 VAR zahl1,zahl2,zahl3:zahl;

{-----------------------------------------------------}
PROCEDURE drucken(z:zahl);
 {Druckt die Zahl z mit fuehrenden Nullen}
 BEGIN
  writeln(z:laenge);
 END;
{------------------------------------------------------}
PROCEDURE eingabe (VAR e:zahl);
 VAR b:zahl;
 BEGIN
  write('?');
  e:=null;
  readln(b);
  e:=copy(e,1,laenge-length(b))
    + copy(b,1,length(b));
 END;
{------------------------------------------------------}
 FUNCTION plus(a,b:zahl):zahl;
  VAR stelle:1..laenge;
      ziffer1,ziffer2,ziffer3:integer;
      code,uebertrag,summe:integer;
      ergebnis:zahl;

   BEGIN
    uebertrag:=0;
    ergebnis:=null;
    FOR stelle:=laenge DOWNTO 1 DO
     BEGIN
      val(a[stelle],ziffer1,code);
      val(b[stelle],ziffer2,code);
      summe:=ziffer1+ziffer2+uebertrag;
      ergebnis[stelle]:=chr(48 +(summe MOD 10));
      uebertrag:=summe DIV 10;
     END;
    plus:=ergebnis;
   END;
{-------- H a u p t p r o g r a m m -----------------}
BEGIN
 writeln('Geben Sie zwei Summanden ein ');
 eingabe(zahl1);
 eingabe(zahl2);
 zahl3:=plus(zahl1,zahl2);
 drucken(zahl1);
 writeln ('+');
 drucken(zahl2);
 writeln ('=');
 drucken(zahl3);
END.
```

Pascal zum Eintippen

Fallen und Fehlerquellen in Pascal: Tabellenausgabe

In der zweiten Folge unserer Reihe, die typische Pascal-Fehler aufzeigt, geht es um Tabellen. Generell betrachtet ist die Ausgabe von Zahlen und Texten in Tabellenform in Pascal ziemlich einfach zu bewerkstelligen. Es gibt bekanntlich die hilfreiche Einrichtung von (optionalen) Formatangaben: nach einem Doppelpunkt wird die Gesamtlänge eines „Fensters" angegeben, in welches der betreffende Wert rechtsbündig eingetragen wird. Beispiel: Wenn die Integer-Variable Summe den Wert 22627 hat, dann bewirkt die Anweisung WRITELN(Summe:8), daß ein „Fenster" mit 8 Stellen ausgegeben wird, wir symbolisieren dies als Kasten:

```
........
```

Der betreffende Wert 22627 wird daher in der folgenden Weise ausgegeben:

```
...22627
```

(Natürlich erscheinen weder der Kasten noch die Punkte tatsächlich in der Ausgabe.)

Sinngemäß gilt dasselbe auch für Texte (Strings) und für Real-Zahlen (siehe auch „Pascal-Lernen mit HC", Teil 2 Heft 5/1986). Bei Real-Zahlen ist außerdem zu beachten, daß die Anzahl der Nachkommastellen durch eine weitere Angabe spezifiziert werden kann. Hat zum Beispiel die Real-Variable x den Wert 61.72, dann führt die Anweisung (WRITELN)(x:8:3) zu der folgenden Ausgabe:

```
..61.720
```

Aber auch mit dieser bequemen Hilfe zum Formatieren können dem Programmierer Fehler unterlaufen. Wir werden uns mit zwei Problemen beschäftigen, die sich häufig stellen.

Problem 1:
Die Rangliste eines Sportclubs soll ausgegeben werden. Die gewünschte Form ist:

Name	Rang
Bauer, A.	8
Bauer, W.	6
Gruber	13
.	.
.	.
.	.
Zechler	9

Wir nehmen an, daß insgesamt 15 Personen zu berücksichtigen sind und daß die Namen alphabetisch geordnet in einem ARRAY[1 . . 15] OF STRING[20] vorliegen, die dazugehörigen Rangplätze sind in einem ARRAY[1 . . 15] OF INTEGER zu finden. Die ARRAY-Variablen heißen Name und Rang.

Es wird behauptet, daß die folgende Anweisungsfolge die gestellte Aufgabe löst:

```
WRITELN('NAME          Rang');
WRITELN('-----------------');
FOR i:=1 TO 15 DO
WRITELN(Name[i]: 20, Rang
[i]:4);
```

Das Ergebnis gleicht jedoch nicht ganz dem Erwünschten. Wieso?

Problem 2:
In einer Tabelle sollen Real-Zahlen in einer Spalte angeordnet werden, wobei das Dezimalkomma tatsächlich als Komma (und nicht als Punkt) zu schreiben ist. Konkreter Fall: In einem ARRAY[1 . .8] OF REAL, das den Namen Betrag hat, sind DM-Beträge gespeichert, die in einer Tabelle der folgenden Form ausgegeben werden sollen:

Tageseinnahmen	
Kasse 1:	606,23 DM
Kasse 2:	461,08 DM
Kasse 3:	509,71 DM
.	.
.	.
.	.
Kasse 8:	495,10 DM

Zur Bearbeitung dieser Aufgabe wird das folgende Programmstück angegeben:

```
WRITELN('Tageseinnahmen');
WRITELN('--------------------');
FOR i:=1 TO 8 DO
BEGIN
  WRITE('Kasse',i:2,':');
  WRITE(TRUNC(Betrag[i]:5,',')
;
  WRITE(ROUND(100*(Bet-
  rag[i]-TRUNC(Betrag[i]))):2);
  WRITELN('DM':3)
END;
```

Auch dieses Programmstück erfüllt die gestellte Aufgabe nicht zufriedenstellend. (Um genau zu sein, es erfüllt sie nur in neun von zehn Fällen.) Wo liegt der Fehler? Die Antworten finden Sie wieder auf der letzten Seite von HC-Extra.

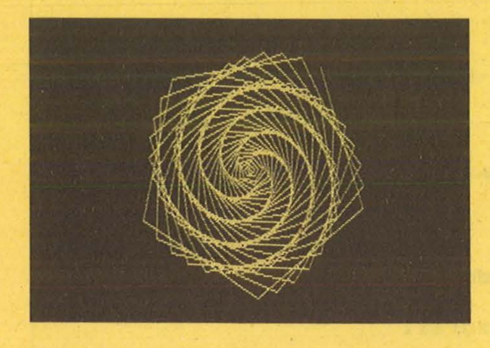
Turtle-Grafik mit Schneider CPC

Schon mit kleinen Programmen kann man durch Veränderung von Parametern eine Vielzahl geometrischer Muster erzeugen. Das folgende Beispiel ist in Turbo-Pascal für Schneider CPC geschrieben und erfordert die eingebundenen Include-Dateien auf der Diskette. Es läßt sich jedoch sehr leicht auf andere Systeme und Sprachen mit Turtle-Grafik übertragen. Dem abgedruckten Programmlisting entnimmt man das einfache Konstruktionsschema. Dazu wird die Schildkröte um eine bestimmte Strecke ('laenge') in der eingestellten Richtung bewegt und anschließend um einen festen Winkel gedreht. Danach wird die eingestellte Länge durch Addition eines festen Wertes 'delta' vergrößert und der Vorgang kann von Neuem beginnen. Die Anzahl der Wiederholungen läßt sich durch den Zahlenwert von 'schritte' steuern. In der folgenden Tabelle sind Werte für sehenswerte Grafiken, wie die auf Seite 63, zusammengestellt.

Drehwinkel	182	60	150	145
Schrittzahl	180	180	180	80
Laenge	0	5	5	180
Laengenänderung	2	1	2	0

Vielfältige Kombinationen

Mit diesem Programm lassen sich auch bekannte geometrische Figuren wie beispielsweise das Quadrat erzeugen. Versuchen Sie es! Schwieriger ist schon der Kreis, den man jedoch mit den Werten



1,360,2,0 erhält. Hinter diesen Zahlen verbirgt sich das Konstruktionsverfahren.

Für besonders experimentierfreudige Programmierer noch ein Tip: Wandeln Sie das Programm in eine Prozedur 'figur' um, der alle bisherigen Parameter sowie die Koordinaten des Anfangspunktes für 'setposition' übergeben werden. Dann kann man durch unterschiedliche Aufrufe von 'figur' ein Gesamtbild aus einzelnen Graphiken konstruieren. Vergessen Sie aber bitte nicht die Beseitigung von 'clearscreen', da ansonsten der Bildschirm nach jeder Einzelzeichnung gelöscht wird. Viel Spaß beim Experimentieren!

Thomas Geise

```
PROGRAM multibild;
{$I grafik3.inc}                    {Enthaelt Grund- und}
{$I turtle3.inc}                    {Turtlegraphik fuer }
                                    {SCHNEIDER CPC      }
   VAR winkel : integer;
        delta : integer;
       laenge : integer;
     schritte : integer;
       nummer : integer;

BEGIN
  write('Drehwinkel ?  '); readln (winkel);
  write('Schrittzahl ? '); readln (schritte);
  write('Startlaenge ? '); readln (laenge );
  write('Laengenaenderung ? '); readln (delta);

  setposition(0,0);               { Bildschirmmitte  }
  clearscreen;                    { Schirm loeschen. }
  for nummer:=1 to schritte do
   BEGIN
    forwd(laenge);                { Geradeaus        }
    turnright(winkel);            { Rechtsdrehung    }
    laenge:=laenge+delta;         { Laengenaenderung }
   END;
END.
```

Mastermind und die Übertragbarkeit von Pascal-Programmen

Eines der Argumente, die als Vorteil des Programmierens in Pascal grundsätzlich angeführt werden, ist die Übertragbarkeit auf andere Rechner. Steht der Anwender dann jedoch vor dieser Aufgabe, so ist von Kompatibilität oft nicht viel zu spüren. Eines der auftretenden Probleme ist die Zuordnung des Dateinamens auf Diskette oder Platte zu einem formalen Namen im Programm. Da Standard-Pascal hierzu keinerlei Mittel bietet, haben die Compiler-Entwickler eigene Lösungen erarbeitet. So fragt MS-Pascal selbständig beim Start des Programmes alle Dateizuordnungen ab, Turbo-Pascal erwartet eine spezielle Anweisung zur Dateizuweisung.

Es gibt aber nach dem Standard von Wirth keine Möglichkeit, diese sehr implementationsabhängigen Lösungen allgemeingültig zu ersetzen, so daß man hier mit Unverträglichkeiten leben muß. Ein weiteres Beispiel zeigt dagegen vermeidbare Inkompatibilitäten, nämlich eine Umdefinition einer festgelegten Standardfunktion: Bei Verwendung der Funktion FRAC kommt es bei manchen Compilern zu Problemen, wenn

```
program mastermind (input,output);

const zeilenlaenge = 40;
       zeilenzahl  = 25;

type zahl        = 0..9;
     zahlfeld    = array[1..4] of zahl;
     bearb←typ   = array[1..4] of boolean;

var  treffer,
     volltreffer   : 0..4;
     versuch,
     loesung       : zahlfeld;
     anz←versuche,
     index:        : integer;
     ver←stelle,
     loe←stelle    : 1..4;
     startwert     : real;
     v←bearbeitet,
     l←bearbeitet  : bearb←typ;

 procedure cls;
  begin
   for index:=1 to zeilenzahl do writeln
  end;

 procedure sternchendrucken;
  begin
   for index:=1 to zeilenlaenge do write('*')
  end;

 function zufall:real;
  var hilfswert:real;
    function frac(a:real):real;
     begin
```

dort anstelle des laut Standard zu liefernden Integerwertes der Compiler ein Real-Ergebnis liefert, da dann unverträgliche Typzuweisungen entstehen. (Falls dieser Fehler auftritt, ist es ratsam, im Handbuch nach einer speziellen Funktion zu suchen, die Realwerte in Integerwerte umwandelt.)
Um Probleme wie die hier beschriebenen zu vermeiden, sollte so weitgehend wie möglich ein dem Standard entsprechender Befehlsatz verwendet werden.
Doch was ist der Standard?
Als Standard kommen im Wesentlichen zwei Festlegungen in Frage: der ursprüngliche Wortschatz, der von Wirth und Jensen im 'PASCAL user manual and report' festgelegt wurde und der neuere ISO-Standard. Im weiteren ist, wenn nicht anders angegeben, die Wirthsche Festlegung gemeint, deren Umfang im Bild dargestellt ist. Es soll an dieser Stelle noch einmal daran erinnert werden, daß jeder Compiler diesen Sprachumfang verarbeiten sollte, jedoch muß er in allen weiteren Worten nicht unbedingt einen Sinn erkennen. Es sei ausdrücklich darauf hingewiesen, daß etwa die im Kurs beschriebenen Funktionen 'length', 'pos' und so weiter Turbo-Pascal-Schöpfungen sind und somit auch nur dort in der beschriebenen Weise wirken.
Der hier diskutierte Standard ist auf den ersten Blick zum Teil recht unbefriedigend, so ist es zum Beispiel lästig, daß Pascal den Datentyp 'string' nicht kennt. Doch man kann ja ebensogut ein ARRAY . . OF CHAR mit entsprechenden Grenzen wählen, etwa 0 . . 80.

Dazu kann man sich die Prozeduren und Funktionen zur Eingabe und Ausgabe selbst schreiben, was allerdings etwas aufwendig ist. Hat man sich jedoch einmal die Mühe gemacht, so kann man diese bei Bedarf leicht in den Quelltext einbauen und das Programm bleibt übertragbar.
Außerdem lernt man beim Erstellen der Prozeduren und Funktionen Pascal außerordentlich gut kennen . . .
Relativ einfach ist die Ausgabe von Zeichenketten zu erreichen, die man als PACKED ARRAY . . OF CHAR deklariert hat. Beispiel:
VAR A: PACKED ARRAY [1 . . 20] OF CHAR; . . .
WRITELN (A);
gibt alle 20 Buchstaben des Arrays aus, wobei sicherzustellen ist, daß diesen auch sinnvolle Werte zugewiesen worden sind. Es ist also nicht notwendig, alle Elemente einzeln durch Nennen der Indices

```
        frac:=a-trunc(a)
       end;
     begin
      hilfswert:=frac(9821*startwert+0.211327);
      startwert:=hilfswert;
      zufall   :=frac(10*hilfswert)
     end;

begin
   cls;
   sternchendrucken;
   writeln('        Z A H L E N R A T E N   0  -  9');
   sternchendrucken;
   writeln;

   writeln('Gib Startwert vor');
   readln (startwert);

   cls;
   for loe←stelle := 1 to 4 do
    begin
      loesung[loe←stelle]:= zufall
    end;

   anz←versuche    := 0;

repeat
        treffer := 0;

  volltreffer := 0;
anz←versuche := anz←versuche + 1;

   for ver←stelle := 1 to 4 do
    begin
      v←bearbeitet[ver←stelle]:=false;
      l←bearbeitet[ver←stelle]:=false
    end;

  writeln;
  writeln('Gib Deinen Versuch ein (0..9) : ');

  for ver←stelle := 1 to 4 do
   begin
    writeln;
    write(ver←stelle,'. Zahl :');
    readln(versuch[ver←stelle])
   end;

  for ver←stelle := 1 to 4 do
   begin
    if (loesung[ver←stelle] =
         versuch[ver←stelle])
     then
      begin
       volltreffer:=volltreffer+1;
       v←bearbeitet [ver←stelle]:=true;
       l←bearbeitet [ver←stelle]:=true
      end
    end;

  for ver←stelle := 1 to 4 do
   for loe←stelle := 1 to 4 do
    begin
     if ((loesung[loe←stelle] =
          versuch[ver←stelle] )
        and (not v←bearbeitet[ver←stelle])
        and (not l←bearbeitet[loe←stelle]))
      then
       begin
        treffer:=treffer+1;
        v←bearbeitet[ver←stelle]:=true;
        l←bearbeitet[loe←stelle]:=true
       end
    end;

 cls;
 sternchendrucken;
```

```
    writeln('     A U S W E R T U N G     ',anz←versuche:9);
    sternchendrucken;
      writeln;writeln;
    for ver←stelle:=1 to 4 do write(versuch[ver←stelle]:3);
      writeln;writeln;
      write('Volltreffer :',volltreffer:2);
      writeln('     Treffer :',treffer:2);

  until volltreffer = 4;

  writeln;
  writeln('Die Zahlenfolge ist richtig .');
  writeln;
  writeln('Du hast ',anz←versuche:3,
                ' Versuche gebraucht !');
  writeln
  end.
```

Sprachumfang von Pascal nach Wirth

Operatoren			
:= div < <> in	+ mod <= and	– / < or	* = <= not
Standardbezeichner			
Konstanten: false	true	maxint	
Typen: integer text	boolean	real	char
Parameter: input	output		
Funktionen: abs arctan chr cos eof	eoln exp ln odd ord	pred round sin sqr sqrt	succ trunc
Prozeduren: get put rewrite	new read unpack	pack readln write	page reset writeln
reservierte Worte			
and array begin case const div do downto else	end file for function goto if in label mod	nil not of or packed procedure program record repeat	set then to type until var while with

auszugeben! Ebenso einfach läßt sich etwa durch mehrfaches Senden eines CHR(10) (=Linefeed) ein Bildschirmlöschen bewirken, ohne daß man auf spezielle Steuerzeichen zurückgreift oder Spracherweiterungen benutzt.

Es ist also durchaus möglich, mit den Sprachmitteln des Standards auszukommen, wenn man nicht gerade Systemprogrammierung im Sinn hat.
Um an einem Beispiel die Umsetzung einer einfachen Aufgabe in Pascal darzustellen, habe ich das Spiel 'Mastermind' gewählt, da es durch einen hohen Bekanntheitsgrad keine weiteren Erläuterungen erfordern dürfte. Es geht dabei darum, Farben in der richtigen Reihenfolge zu erraten. Dabei habe ich konsequent alle Spracherweiterungen vermieden, um zu zeigen, daß es auch ohne spezielle Tricks möglich ist, sich die entsprechenden Routinen selbst zu schreiben.
Der Algorithmus zur Erzeugung der Zufallszahlen wurde der HP65 Anwenderbibliothek entnommen, er stammt von Don Malm und sei zur weiteren Verwendung wärmstens empfohlen.
Das dabei verwendete Verfahren erfordert die Eingabe eines Startwertes für die Zufallszahl zwischen 0 und 1, alle weiteren 'Zufallszahlen' werden daraus abgeleitet; es sollte daher nicht mit 0 begonnen werden. *Volker Mühlhaus*

Rechnen leicht gemacht

Es ist schon etwas aberwitzig, einen Computer zu benutzen und für Rechenoperationen dann doch zum Taschenrechner zu greifen. Ist es nicht sinnvoller, sich ein kleines Taschenrechner-Programm zu schreiben und dieses fortan zum Addieren wie zum Wurzelziehen zu verwenden?
Die Antwort ist ein eindeutiges Nein. Wer auf Rechengenauigkeit auch bei sehr großen oder kleinen Zahlen Wert legt, dem bleibt gar nichts anderes übrig, als sich auf einen hochwertigen Taschenrechner zu verlassen. Eine vergleichbar hohe Rechengenauigkeit kann bei Home-Computern nur mit programmiertechnischen Klimmzügen erreicht werden. Außerdem macht sich der eingeschränkte Rechenbereich unangenehm bemerkbar. Die meisten Taschenrechner erfassen Zahlen mit bis zu 99 Stellen, beim Computer sind es ohne große Tricks gerade ein Drittel soviel.
Ein anderer Punkt ist die Geschwindigkeit. Auch in diesem Punkt ist der Fachidiot Taschenrechner dem heimischen Computer meist überlegen.
Das heißt aber nicht, daß die Rechenmaschine Computer ihre ureigenste Aufgabe nicht erfüllt und im Rechnen schwach ist. Ein Beweis ist das vorliegende Programm, dessen Geschwindigkeit in erster Linie von der Schnelligkeit des Bildschirmaufbaus abhängt, sofern es wie geschehen unter Turbo-Pascal eingesetzt wird.

Die Aufgabenstellung: Mit möglichst wenig Aufwand sollen die Grundfunktionen eines sich selbst dokumentierenden Taschenrechners simuliert werden. Die Eingabe hat dabei dem üblichen Muster zu folgen. Mit einer kleinen Ausnahme erfüllt das Programm diese Auflagen: Nach Eingabe der ersten Zahl muß vor der gewünschten Operation <RETURN> gedrückt werden, außerdem hat <RETURN> das Gleichheitszeichen ersetzt. Zum Beispiel muß 2 plus 2 eingegeben werden wie: 2<RETURN>+2<RETURN>
Mit dem Ergebnis kann dann wie gewohnt weitergerechnet werden. Auch weitergehende Funktionen, wie die Bildung von Quadraten, sind ohne weiteres möglich, wenn als Operation das entsprechende Buchstabenkürzel – etwa q für quadrieren – eingegeben und mit <RETURN> als Gleichheitszeichen quittiert wird.
Die Funktionen des Programms entsprechen dabei denen von Standard-Pascal, wie ein Blick auf die Tabelle auf Seite 65 beweist. Nicht dem Standard entspricht CLRSCR; diese Prozedur kann aber problemlos eingebaut werden:

```
procedure clrscr;
begin
   write <chr<12> >;
end;
```

Im einzelnen verfügt das Programm über Funktionen zum Quadrieren <q>, Wurzelziehen <w>, Potenzieren <e> und trigonometrische Grundfunktionen im Bogenmaß. *-dw*

```
PROGRAM taschenrechner;    {HC-EXTRA, Dieter Winkler, 7.7.86}

  VAR     zahl1,zahl2 : real;
            ergebnis : real;
                   k : char;

PROCEDURE abstand;
  BEGIN;
    writeln; writeln; writeln; writeln; writeln;
  END;

BEGIN
  clrscr;
   writeln ('        Simulation eines Taschenrechners');
   writeln ('        ================================');
   writeln;
   writeln ('Zahlen jeweils abschliessen mit:    (RETURN)');
   write ('Der Druck auf die (RETURN)-Taste ersetzt dabei');
   writeln (' das Gleichheitszeichen');
      abstand;
       write('1. Zahl eingeben              ');readln(zahl1);
         ergebnis := zahl1;
                    REPEAT                      {Rechenschleife}
         zahl2 := 0;
      abstand;
       write('Operation eingeben          ');
       read(kbd,k);
       write(k);
      abstand;
       write('2. Zahl eingeben            ');readln(zahl2);
            IF k = '+' THEN ergebnis := zahl1 + zahl2;
            IF k = '-' THEN ergebnis := zahl1 - zahl2;
            IF k = '*' THEN ergebnis := zahl1 * zahl2;
            IF k = ':' THEN ergebnis := zahl1 / zahl2;
            IF k = 'q' THEN ergebnis := SQR(ergebnis);
            IF k = 'w' THEN ergebnis := SQRT(ergebnis);
            IF k = 's' THEN ergebnis := SIN(ergebnis);
            IF k = 'c' THEN ergebnis := COS(ergebnis);
            IF k = 'a' THEN ergebnis := ARCTAN(ergebnis);
            IF k = 'e' THEN ergebnis := EXP(ergebnis);
            IF k = 'l' THEN ergebnis := LN(ergebnis);
  clrscr;
    writeln ('Fuer Ende als Operation eingeben:  x (RETURN)');
    writeln ('Neustart:                          r');
      abstand; abstand;
    write (zahl1:8:4,' ',k,zahl2:8:4,' =',ergebnis:8:4);
         zahl1:=ergebnis;
                    UNTIL k='x'
END.
```

Lösungen zu S. 62

Lösung zu Problem 1: Die Anweisungsfolge führt zu einer Tabelle der Gestalt

Name	Rang
Bauer,A.	8
Bauer,W.	6
Gruber	13
.	.
.	.
.	.
Zechler	9

Wir sehen, daß die Namen rechtsbündig in 20 Stellen breite Fenster gesetzt werden, gefordert ist aber die linksbündige Anordnung. Um diese zu erzielen, kann jeweils der Name ohne Formatangabe ausgegeben werden, gefolgt von einer variablen Anzahl von Leerstellen. Deren Anzahl ist 20 minus der Länge des Namens. Eine Lösung des gestellten Problems ist somit das folgende Programmstück:

```
WRITELN('Name              Rang
');WRITELN('-------------------');
FOR i:=1 TO 15 DO
  WRITELN
(Name[i],'':20-LENGTH(Name[i]
,,RANG[i]:4);
```

Lösung zu Problem 2:
In dem angegebenen Programmstück liegt ein subtiler Fehler versteckt. Die Anweisungsfolge ist nur ausreichend für alle Zahlen, deren Pfennigbetrag zweistellig ist. Wenn jedoch z.B. 461,08 auszugeben ist, erscheint statt dessen 461, 8. Eine Lösung dieser Schwierigkeit besteht darin, solche Fälle getrennt zu berücksichtigen und die fehlende 0 direkt zu ergänzen. Dies geschieht im folgenden Programmstück, wobei der Übersichtlichkeit halber zwei Hilfsvariable Mark und Pf vom Typ Integer benutzt werden:

```
WRITELN('Tageseinnahmen');
WRITELN('-------------------');
FOR i:=1 TO 8 DO
BEGIN
  Mark:=TRUNC(Betrag[i];
  Pf:=ROUND(100*(Betrag[i]-
  Mark))
  WRITE('Kasse',i:2,':');
  WRITE(Mark:5);
  IF Pf<10 THEN
  WRITE('0',Pf:1)
          ELSE WRITE(Pf:2);
  WRITELN('DM':3)
END;
```

Riesige Primzahl gefunden

Die große Beteiligung am Primzahl-Wettbewerb aus HC 5/86 hat gezeigt, daß wir mit unserem Vorhaben richtig liegen, den Lesern regelmäßig Aufgaben zu stellen. Mit dem Einsendeschluß am 15. Mai 1986 standen maximal etwa vier Wochen zur Entwicklung des Programmes und zur Bestimmung der Primzahl durch das Programm zur Verfügung. Wir hatten eigentlich vermutet, daß das Programm des Gewinners an diese Zeitgrenze stoßen würde, also eine noch größere Primzahl gefunden hätte, wenn mehr Zeit zur Verfügung gewesen wäre.

Es kam aber anders: die größte Primzahl belegte mit 16 354 Ziffern fast den ganzen Speicher eines C64. Sie wurde von Joachim Pfefferle aus Balingen ermittelt, der mit dieser Leistung sich den Gewinn, einen Epson LX-90-Drucker, redlich verdient hat. Dazu brauchte ein Maschinenprogramm nur zirka neun Stunden. Bei einem größeren Speicher hätte dieser Leser mit dem gleichen Programm leicht eine noch größere Primzahl ermitteln können. Allerdings muß gesagt werden, daß mit einem weiteren Trick (Speicherung von zwei Ziffern in einem Byte) auch der Speicher des C64 für weitaus größere Primzahlen ausgereicht hätte. Der Gewinner räumt dies und weitere Optimierungsmöglichkeiten selbst ein.

Berücksichtigt wurden Einsendungen nicht, bei denen das Programm selbst gar nicht garantiert, ob die errechnete Zahl eine Primzahl ist. Wenn also zum Beispiel vorausgesetzt wurde, daß $2^{132049}-1$ eine Primzahl ist (was man in etlichen Büchern bestätigt finden kann) und das Programm dann nur unzählige Verdopplungen vornimmt und zum Schluß eins subtrahiert, dann war das nicht im Sinne des Aufrufs in HC 5/86. Trotzdem ist auch diese Programmierleistung nicht ganz selbstverständlich, denn auch hier mußte (wegen der Größe der Zahl) vom vorgegebenen Integer-Format des Rechners abgewichen werden und (aus Zeitgründen) in Maschinensprache programmiert werden. Doch lassen wir jetzt den Gewinner mit seiner Lösungsmethode zu Worte kommen. *-br*

Das Gewinner-Programm

Bedienung

Nach dem Laden und Starten wird sofort ein MC-Programm nachgeladen. Es erscheint dann eine Frage: Hier können Sie eine Zahl kleiner als die maximal mögliche eingeben und das Programm gibt dann die Primzahl mit der nächstgrößeren Zahl von Ziffern aus. Beispiel: Eingabe 100 – es erscheint eine Primzahl mit 101 Ziffern. Durch Drücken der „P"-Taste kann das Programm unterbrochen und die aktuelle Primzahl ausgegeben werden. Erneutes Drücken einer Taste läßt das Programm fortfahren. Wenn bei der Ausgabe die Primzahl eine gewisse Länge überschreitet, wird die Ausgabe unterbrochen. Fortsetzung durch Drücken der Leertaste. Um die größte Primzahl vom Programm ausgeben zu lassen, geben Sie eine Zahl größer oder gleich 16354 ein.

Theorie

Das Prinzip ist schnell erklärt: Wenn man die ersten n Primzahlen miteinander multipliziert und dazu 1 addiert, so ist diese neue Zahl durch keine andere außer sich selbst und 1 teilbar, also eine neue Primzahl. Beispiel: $2 \times 3 \times 5 + 1 = 31$ ist eine neue Primzahl. Jetzt braucht man nur noch

die ersten n Primzahlen, die wie folgt erzeugt werden: Mit 3 beginnend, wird jede ungerade Zahl daraufhin überprüft, ob sie durch eine Primzahl teilbar ist. Ist das nicht der Fall, so handelt es sich um eine Primzahl.

Zum Programm

Es besteht hauptsächlich aus drei MC-Routinen: der Multiplikationsroutine, der Ausgaberoutine und dem Programm zur Berechnung der nächsten kleinen Primzahl (siehe Theorie). Die kleine Primzahl-Routine beginnt bei $C810. Die kleinen Primzahlen werden als 2-Byte-Integers ab Adresse $6000 gespeichert. Die Multiplikationsroutine multipliziert nun die kleine Primzahl Ziffer für Ziffer mit der großen. Nach dem Addieren, Übertrag addieren etc. steht dann das Ergebnis anstelle der alten Primzahl im Speicher (ab $1604 = letzte Ziffer, pro Byte eine Ziffer). Für eine Primzahl muß noch 1 zur letzten Ziffer addiert werden. Das erfolgt bei der Ausgabe. Im Speicher bleibt also das Multiplikationsergebnis.

Noch etwas zum Programm: Es mußte leider mit einem Monitorprogramm programmiert werden. Es gleicht daher auch eher einem mittleren Chaos als einem gut strukturierten Programm.

Zeitdauer für 1000 Stellen ca. 3 Minuten
Zeitdauer für 7000 Stellen ca. 2,5 Stunden
Zeitdauer für 16 000 Stellen ca. 9,0 Stunden

Größte Primzahl 16354 Ziffern ab $1604 (letzte Ziffer) im Speicher, pro Ziffer 1 Byte.

Leider konnte das RAM unter dem BASIC-ROM nicht genutzt werden – das Monitorprogramm hat da nicht mitgespielt. Sonst wäre eine noch beträchtlich größere Anzahl von Ziffern möglich gewesen, allerdings nur mit sehr viel Geduld.

Druckerausgabe

Von BASIC aus kann man die Primzahl so ausdrucken:

T = 256 * PEEK (49787) + PEEK (49786)
FOR X = T TO 5637 STEP – 1
PRINT PEEK (X); = NEXT
PRINT PEEK (5636) + 1 : REM = Letzte Ziffer + 1

Wenn man nun vorher den Kanal zum Drucker öffnet mit
OPEN 1,4 : CMD 1
und hinterher mittels
PRINT#1 : CLOSE 1
wieder schließt, müßte man die Zahl auf dem Drucker ausgeben können (siehe Programm).

Im Programm kann man die Primzahl, nachdem sie auf dem Bildschirm ausgedruckt wurde, durch Drücken der „D"-Taste auf dem Drucker ausgeben.

Joachim Pfefferle

Die neueste Programm-Version

Die maximale Stellenzahl beträgt jetzt etwa 28 280 Stellen. Wir haben diese anstelle des Gewinner-Programms hier abgedruckt. Ein kommentiertes Assembler-Listing und eine aktualisierte Beschreibung folgen im nächsten Heft.

Joachim Pfefferle

```
1 GOTO100
5 :
15 :
20 :      EIN PRIMZAHLENPROGRAMM VON
30 :           JOACHIM PFEFFERLE
40 :           ERZINGER WEG 2
50 :           7460 BALINGEN
60 :                5/86
70 :
80 :
100 REM BASICENDE HERABSETZEN
110 POKE55,255:POKE56,21:POKE51,255:POKE
52,21
120 :
130 REM  EVENT.DATA LESEN
140 IFPEEK(49152)=169ANDPEEK(49413)=124G
OTO200
150 FORX=49152TO49637:READW:POKEX,W :SU=
SU+W :NEXT X
160 IFSU<>67510THENPRINT"FEHLER IN DATAS
!":END
190 :
200 POKE53280,11:REM  FARBEN SETZEN
210 POKE53281,6
220 POKE646,13
230 :
300 INPUT"ANZAHL DER STELLEN";S
310 SYS49152:POKE249,0:POKE250,22
330 PRINT"■■"; "STELLEN           KLEINE
 PRIMZ."
335 :
340 SYS49218 :REM HAUPTROUTINE
450 T=256*PEEK(49787)+PEEK(49786):LA=T-5
635
550 KP=256*PEEK(49789)+PEEK(49788)
560 IFLA>=SORKP>65510THEN1000
570 PRINT"■";LA;"                   ";KP
580 :
600 GETA$:IFA$<>"P"THEN 680
610 PRINT"■■■":SYS49576:REM AUSGABE
640 GETA$:IFA$=""ORA$=" "THEN640
650 IFA$="D"THEN GOSUB 2000:GOTO640
670 PRINT"■■STELLEN                KLEINE PR
IMZ."
680 POKE249,0:POKE250,22
700 GOTO340
710 :
1000 PRINT"■":PRINT:PRINT"■":SYS49576
1010 PRINT: PRINTLA;"STELLEN"
1020 GETA$:IFA$=""THEN1020
1030 IFA$="D"THEN GOSUB 2000:GOTO1020
1040 END
2000 OPEN1,4:CMD1
2010 FORX=T TO 5637 STEP-1 :PRINTRIGHT$(
STR$(PEEK(X)),1);:NEXT
2020 PRINTRIGHT$(STR$(PEEK(5636)+1),1)
2030 PRINT#1:CLOSE1
2050 RETURN
5000 DATA169,22,133,250,169,0,133,249
5002 DATA160,0,145,249,230,249,208,250
5004 DATA230,250,165,250,201,160,208,236
5006 DATA169,0,133,249,169,22,133,250
5008 DATA169,0,162,32,157,105,194,202
5010 DATA208,250,169,22,141,123,194,169
5012 DATA4,141,122,194,169,3,141,132
5014 DATA194,141,124,194,169,2,141,4
5016 DATA22,234,120,160,4,162,0,189
5018 DATA128,194,240,19,152,24,109,122
5020 DATA194,141,122,194,169,0,109,123
5022 DATA194,141,123,194,76,99,192,232
5024 DATA136,208,228,160,0,177,249,141
5026 DATA118,194,190,128,194,248,202,48
5028 DATA33,240,24,24,109,118,194,202
5030 DATA208,249,170,41,240,74,74,74
5032 DATA74,24,109,117,194,141,117,194
5034 DATA138,41,15,24,109,112,194,141
5036 DATA112,194,216,200,192,5,208,205
5038 DATA173,112,194,170,41,240,74,74
5040 DATA74,74,248,24,109,117,194,141
5042 DATA117,194,216,138,41,15,141,112
5044 DATA194,173,116,194,160,0,145,249
5046 DATA173,115,194,141,116,194,173,114
5048 DATA194,141,115,194,173,113,194,141
5050 DATA114,194,173,112,194,141,113,194
5052 DATA173,117,194,141,112,194,162,0
5054 DATA142,117,194,165,249,205,122,194
5056 DATA240,9,230,249,208,2,230,250
5058 DATA76,99,192,165,250,205,123,194
5060 DATA208,240,173,116,194,240,12,238
5062 DATA122,194,208,3,238,123,194,76
5064 DATA226,192,234,24,173,124,194,105
5066 DATA2,141,124,194,173,125,194,105
5068 DATA0,141,125,194,169,0,141,1
5070 DATA195,169,3,141,0,195,173,0
```

```
5072 DATA195,133,89,173,1,195,133,90
5074 DATA173,124,194,133,87,173,125,194
5076 DATA133,88,32,129,193,165,92,208
5078 DATA4,165,93,240,198,165,88,208
5080 DATA6,165,87,201,3,144,20,24
5082 DATA173,0,195,105,2,141,0,195
5084 DATA173,1,195,105,0,141,1,195
5086 DATA76,30,193,173,124,194,133,87
5088 DATA173,125,194,133,88,169,10,133
5090 DATA89,169,0,133,90,162,4,138
5092 DATA72,32,129,193,104,170,165,92
5094 DATA157,128,194,202,16,241,88,96
5096 DATA234,162,0,134,92,134,93,160
5098 DATA16,6,87,38,88,38,92,38
5100 DATA93,56,165,92,229,89,170,165
5102 DATA93,229,90,144,6,134,92,133
5104 DATA93,230,87,136,208,227,96,234
5106 DATA165,249,56,233,200,141,119,194
5108 DATA160,0,177,249,166,249,224,4
5110 DATA208,6,166,250,224,22,240,31
5112 DATA24,105,48,32,210,255,165,249
5114 DATA208,2,198,250,198,249,173,119
5116 DATA194,197,249,208,221,32,225,255
5118 DATA201,239,208,249,76,178,193,24
5120 DATA105,49,32,210,255,96
READY.
```

Bit-Parade

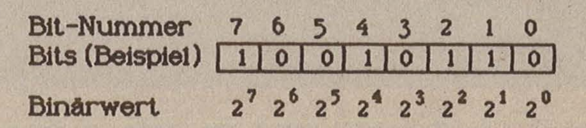
Nichts als Zahlen kennt der Computer oder genauer: nur die Zustände Aus (0) und Ein (1). Daß man trotzdem mit ihm rechnen kann, wird klar, wenn man eine Handvoll Fachchinesisch verstanden hat. Hier die wichtigsten Begriffe (Teil 1).

Adresse
Das Gedächtnis des Rechners, der Speicher, besteht aus einer Unzahl von Schaltelementen, die entweder ein- oder ausgeschaltet sein können. Mehrere dieser Elemente (→ Bit), bei Home-Computern sind es acht, sind zu einer Einheit (→ Byte) zusammengefaßt und über eine fortlaufende Nummer wie ein Haus in einer Straße auffindbar, die Adresse.
Schluckt der Prozessor des Computers jeweils acht Bit am Stück und verwendet er zweimal acht Bit für die Adressierung, kann er einen Adreßraum von 65536 Speicherzellen verwalten. Wieso? 8 bit können von 0 bis 255 zählen, 256 ∗ 256 = 65536. Dieser Wert entspricht 64 Kilobyte (KByte) und damit ist dann auch klar, warum Home-Computer hier die Grenzen ihres Wachstums finden.

ASCII
Mit dem Fernschreiber ging es los. Um Text zu übertragen, mußten die Schriftzeichen in elektrische Impulse umgewandelt werden. Zu diesem Zweck ordnete man jedem Buchstaben, jeder Ziffer, jedem Zeichen, aber auch den Steuerfunktionen wie Papiervorschub, Glokkensignal etc. einen Zahlenwert zu.
Damit sich Fernschreiber verschiedener Hersteller verstehen konnten, mußte man sich auf einen Standard einigen, den **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange. Danach hat das „A" zum Beispiel den Code 65, die Leerstelle hat den Code 32.
Computer arbeiteten anfangs mit Fernschreibern als Ein- und Ausgabegerät. So fand der ASCII-Code Eingang in die Elektronenhirne. Ursprünglich nur für 128 Zeichen definiert (7 bit ASCII), werden heute meist 256 Characters (8 bit) bestimmt. Ohne einheitliche Norm geht es da zumindest bei den Blockgrafikzeichen wieder drunter und rüber. Und weil ASCII die Kommunikation zwischen Rechner und Drucker erst ermöglicht, können dann schwarz auf weiß Probleme auftauchen.

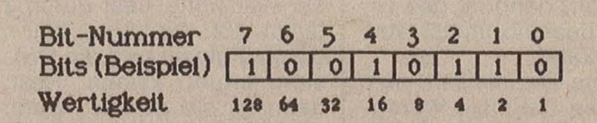
BCD-Konstanten
Die Abkürzung heißt ausgeschrieben: **B**inary **C**oded **D**ecimal, binär codierte Dezimale und bezeichnet eine Methode, um Dezimalzahlen mit den Möglichkeiten des Rechners darzustellen.
Statt die Dezimalzahl in ihr binäres Äquivalent umzurechnen, wird jede Ziffer der Dezimalzahl für sich in einen Binärwert gewandelt. Um Ziffer von 0 bis 9 darstellen zu können, werden vier Bit benötigt. In jeder 8-bit-Adresse können so zwei Dezimalziffern gespeichert werden. In einem weiteren Byte werden das Vorzeichen (+/−) und der Exponent (Stellen vor und hinter dem Komma) abgelegt.
Die Länge der BCD-Konstanten bestimmt die Rechengenauigkeit. Bei 4 Byte BCD können in drei Byte insgesamt sechs Dezimalstellen erfaßt werden (einfache Genauigkeit), bei 8 Byte BCD (doppelte Genauigkeit) sind es schon vierzehn.

Binär
Binär oder dual heißt soviel wie „zweihaft". Mathematiker bezeichnen damit ein Zahlensystem, das auf die Basis 2 bezogen ist, d.h. jede Stelle innerhalb einer Binärzahl entspricht einer Zweierpotenz und es werden nur zwei Ziffern geschrieben.
Dieses Zahlensystem ermöglicht dem Rechner mathematische Operatoren mit nur zwei Spannungszuständen, Aus = 0 und Ein = 1, durchzuführen.



Bit
Binary digit, Binärziffer oder -stelle, bezeichnet ein Speicherelement, das zwei Zustände annehmen kann: Aus (0) oder Ein (1). Bit ist die kleinste Maßeinheit für Information, eine Ja/Nein-Entscheidung.

Byte
Die anfänglich für verschieden große Zusammenfassungen von Bit verwendete Bezeichnung **binary trice** ist zu einer festen Maßeinheit geworden: 8 bit = 1 Byte.
Die heute noch üblichen 8-bit-Prozessoren verwalten Speicheradressen von je 8 bit, 1 Byte ist also die kleinste adressierbare Einheit, sie kann dezimale Werte von 0 bis 255 annehmen (→ Adresse).



Dezimal
Unsere vertraute Art, Zahlen zu notieren, ist das Dezimalsystem. Es werden zehn Ziffern von 0 bis 9 verwendet, die Stellung innerhalb der Zahl gibt der Ziffer den Stellenwert, eine entsprechende Zehnerpotenz (1, 10, 100 usw.), die Ziffer bezeichnet den Faktor, also: 7351 meint
7 ∗ 10hoch3 (1000) +
3 ∗ 10hoch2 (100) +
5 ∗ 10hoch1 (10) +
1 ∗ 10hoch0 (1). Entsprechend werden andere Zahlensysteme aufgebaut, nur die Basis, beim Dezimalsystem 10, ändert sich.

Flag
Um bestimmte Zustände zu kennzeichnen, zum Beispiel ein angeschlossenes Peripheriegerät, einen aktivierten Bildschirmmodus, den Zustand eines Steuerknüppels (Joystick) werden in bestimmten Speicherplätzen einzelne Bit gesetzt, also auf „1" geschaltet.
So kann beispielsweise durch das Setzen von Bit 4 in Adresse 756 die doppelte Größe für Sprites eingeschaltet werden. Eine solche Markierung heißt Flag.

Gesetzt
Ein Bit ist gesetzt, wenn es den Wert 1 hat. Von BASIC aus ist das aber nicht ohne weiteres zu ermitteln, denn die kleinste adressierbare Einheit ist ein Byte, bestehend aus acht Bit.

AKTIV COMPUTERN

Ist Bit 4 in Adresse 756 gesetzt? Der Inhalt von 756 wird in dezimaler Form ausgegeben: enthält zum Beispiel der Dezimalwert 186 ein gesetztes viertes Bit? Um diese Frage zu beantworten, muß BASIC den Dezimalwert (D) aufdröseln. Das kann je nach Dialekt etwa so aussehen:

```
1000 FOR J=0 TO 7:B(J)=0:NEXT J
1010 IF D>127 THEN B(7)=1:D=D-128
1020 IF D>63 THEN B(6)=1:D=D-64
1030 IF D>31 THEN B(5)=1:D=D-32
1040 IF D>15 THEN B(4)=1:D=D-16
1050 IF D>7 THEN B(3)=1:D=D-8
1060 IF D>3 THEN B(2)=1:D=D-4
1070 IF D»1 THEN B(1)=1:D=D-2
1080 B(0)=D
```

Durch diese Routine wird jeder Dezimalwert (von 0 bis 255) in seine acht Bit (B(7) bis B(0)) zerlegt. Natürlich kann man die acht Zeilen auch in eine FOR-NEXT-Schleife packen, aber die dann nötigen vielen Potenzen machen das Programm letztlich langsamer:

```
1000 FOR J=7 TO 0 STEP -1
1010 B(J)=0:W=2^J
1020 IF D>W-1 THEN B(J)=1
:D=D-W
1030 NEXT J
```

Wenn aber nur der Zustand eines einzigen Bit gefragt ist, kann der durch eine Formel überprüft werden. Um diese Formel zu verstehen, muß man sich das Verhältnis der dezimalen Wertigkeiten der einzelnen Bit deutlich machen.

Die Wertigkeit von Bit n ist genau doppelt so groß wie die von Bit n−1. Wird die Wertigkeit irgendeines höheren Bit durch die Wertigkeit irgendeines niedrigeren Bit geteilt, dann ist das Ergebnis deshalb immer eine gerade Zahl.

Wird eine Dezimalzahl also durch die dezimale Wertigkeit eines Bit geteilt und das Ergebnis ist eine gerade Zahl, dann ist nicht dieses, sondern irgendein höheres Bit gesetzt. Nur wenn das Ergebnis ungerade, nämlich 1, ist, dann ist genau das gefragte Bit gesetzt.

Die Formel muß also folgende Arbeitsschritte erledigen:

1: die Wertigkeit des Xten Bit berechnen (= 2hochX);
2: den Dezimalwert D durch diese Wertigkeit teilen;
3: von diesem Quotienten nur den ganzzahligen Teil (INT) berücksichtigen;
4: feststellen, ob dieser Wert gerade ist.

Eine Zahl ist gerade, wenn sie sich ohne Rest durch 2 teilen läßt. Wird eine Zahl n durch 2 geteilt und die Division ergibt keinen Rest, dann ist sie gerade. Wenn also der Quotient n/2 gleich der Ganzzahl INT (n/2) ist, dann ist n eine gerade Zahl (ist doch klar, oder?).

Die vollständige Formel lautet:

```
INT(INT(D/2^X)/2)<>INT(D/
2^X)/2
```

IF diese Formel erfüllt THEN Bit X=1 (gesetzt).

```
10 ? "In welcher Adresse
liegt das gesuchte Bit?":
?
20 INPUT A:D=PEEK(A):? :?

30 ? "Den Zustand von wel
chem Bit (7 bis 0) wollen
 Sie wissen?":?
40 INPUT X
50 IF INT(INT(D/2^X)/2)<>
INT(D/2^X)/2 THEN B=1
60 ? :? "In der Adresse "
;A;" ist Bit #";X;"=";B:?
:? :? :GOTO 10
```

Hex
Das Hexadezimalsystem notiert Zahlenwerte zur Basis 16. Es werden sechzehn Ziffern benötigt: 0 bis 9, A, B, C, D, E und F. Jede Stelle in einer Hex-Zahl bedeutet eine 16er Potenz; also: 3D1A entspricht 3 ∗ 16hoch3 + 13 ∗ 16hoch2 + 1 ∗ 16hoch1 + 10 ∗ 16hoch0 = 3 ∗ 4096 + 13 ∗ 256 + 1 ∗ 16 + 10 ∗ 1 = 15642, dezimal versteht sich.

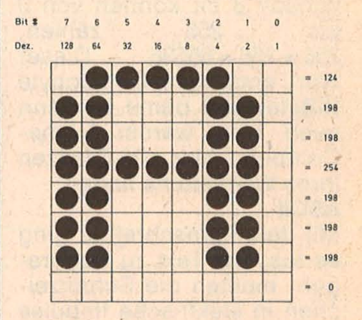
Invertieren
Warum einfach, wenn's auch ausländisch geht? Invertieren heißt umdrehen, gesetzte Bit (1) sollen nicht gesetzt (0) werden und umgekehrt.
Wenn Schriftzeichen negativ auf dem Bildschirm erscheinen sollen, dann werden die Daten, die sie zur Darstellung bringen, invertiert.
Beispiel: Ein Byte hat den Inhalt 01101101, der Rechner gibt das aber nur dezimal aus: 109. Wird das Byte invertiert, soll es den binären Inhalt 10010010 bekommen. Der ursprüngliche Binärwert des Byte und der invertierte Wert ergeben addiert 11111111 oder dezimal 255.
109 + DI muß also 255 ergeben oder 255 − 109 ergibt den Dezimalwert (DI) des invertierten Byte, nämlich 146 dezimal und das ist doch tatsächlich 10010010, binär.

Kilo-Byte
Die Bezeichnung Kilo meint üblicherweise 1000 Kilogramm: 1000 Gramm. Weil die Computertechnik nun aber einmal vom Dualsystem regiert wird und 2hoch10 nun mal 1024 ist, bezeichnet ein Kilobyte (KByte) nun mal nicht 1000, sondern 1024 Byte.

Maske
Bit-Masken werden auf der Ebene von Maschinensprache verwendet, um festzustellen, ob bestimmte Bit gesetzt sind. Der Inhalt einer Speicherzelle, z.B. 10010110, wird mit der Bit-Maske z.B, 00110000, durch das logische AND verknüpft. Das Ergebnis ist ein Wert, bei dem nur die Bit „1" sind, die sowohl in der Speicherzelle als auch in der Bit-Maske gesetzt sind:
Zelle 10010110
Maske 00110000
AND
00010000
Die Bit-Masken-Technik funktioniert nur, wenn Bit für Bit durch AND verknüpft wird. Deshalb ist sie aus BASIC heraus nicht anwendbar, denn BASIC reicht nur an Byte nicht an Bit heran.

Mega-Byte
Die Vorsilben Mega bezeichnen bei Maßangaben den Wert Million. Ein Megawatt entspricht einer Million Watt, so wie Kilowatt tausend Watt bedeutet.
Da das Informationsmaß KByte nicht 1000, sondern 1024 Byte bezeichnet, ist 1 MByte entsprechend 1 048 576, nämlich 1024 × 1024 Byte.

Muster
Unter Bit-Muster versteht man die Anordnung gesetzter und nicht gesetzter Bit innerhalb eines Byte. Ersetzt man gesetzte Bit durch leuchtende Punkte auf dem Bildschirm oder druckende Punkte beim Nadeldrucker, wird der Name restlos verständlich.
Jeder dezimale Werte repräsentiert ein Bit-Muster. Mit acht Byte kann eine Bildfläche von 8 × 8 Punkten erfaßt werden. Die Darstellung eines Zeichens auf dem Bildschirm benötigt deshalb acht Byte:



Nibble
Ein Byte umfaßt acht Bit. Ein halbes Byte, also vier Bit, bezeichnet man als Nibble. Der Begriff hat seine Bedeutung, weil vier Bit genau einer Hex-Ziffer entsprechen:
dezimal 0110 1100
hex 6 C

Der zweite und letzte Teil der Bit-Parade folgt in der nächsten HC-Ausgabe.

Karl-Heinz Koch

Z80-Betriebssystem im Eigenbau

Teil 4: Ausgabe von Zahlen für Schneider CPC und Sinclair Spectrum

Ein eigenes Betriebssystem ohne Zugriff auf das Original-ROM muß auch eine Routine aufweisen, die den Inhalt von Zahlvariablen auf den Bildschirm bringt.

Sobald Z80-Maschinenprogramme mit Zahlvariablen umgehen, die später Ziffer für Ziffer auf den Bildschirm ausgegeben werden sollen, ist es vorteilhaft, sie im BCD-Code abzuspeichern. Beim BCD-Code wird in den oberen vier Bit und den unteren vier Bit eines Byte jeweils eine Ziffer (0 bis 9) abgelegt. Mit vier Bit kann man eigentlich die Zahl 0 bis 15 darstellen, 10 bis 15 entfallen jedoch im BCD-Code. Rechnungen im BCD-Code sind etwas umständlich, denn sie müssen stets im Akkumulator ausgeführt werden, weil die erneute Anpassung an das BCD-Format mit dem Befehl DAA nur Wirkung auf den Akkummulator hat.

Das Programm ZIFF bringt eine ganze Zahl mit zwei Ziffern in die Zeile D ab der Spalte E des Bildschirms. HL zeigt beim Aufruf von ZIFF auf die Speicherstelle, an der diese Zahl im BCD-Code steht. Soll die Zahl vierfach vergrößert erscheinen, dann muß Bit 7 von D gesetzt sein. ZIFF legt bei STRG+2 das Zeichen # ab und schreibt die beiden Ziffern einzeln in STRG bzw. STRG+1. Damit ist die eigentliche Zifferndarstellung für das Programm TEX durchführbar, mit HL als Textzeiger auf STRG.

ZIFF ruft das Programm TEX auf, TEX wiederum benötigt die Routine ZWIL. Beide müssen also im Speicher sein. TEX und ZWIL wurden in den letzten Folgen dieser Serie vorgestellt. In TEX für Sinclair Spectrum muß die letzte Anweisung, V DEFB 73, weggelassen werden, weil V bereits im Hauptprogramm zum Aufruf von ZIFF festgelegt wird.

In der nächsten Folge sollen unter anderem Routinen zum Rechnen im BCD-Code ausführlich besprochen werden. *-br*

```
;                 ZIFFERNDARSTELLUNG
;
ZIFF    PUSH      HL           ;Der Zeiger auf die Zahl wird gerettet.
        LD        A,35         ;Zur Markierung des Textendes für TEX
        LD        (STRG+2),A   ;kommt das Zeichen # nach STRG+2.
        LD        A,(HL)       ;Von der doppelziffrigen Zahl
        SRL       A            ;wird die höherwertige Ziffer aus
        SRL       A            ;der oberen Hälfte des Bytes in
        SRL       A            ;die untere Hälfte geschoben
        SRL       A            ;(der Rest ist mit Nullen aufgefüllt).
        ADD       48           ;Addition der ASCII-Codezahl von "0".
        LD        (STRG),A     ;Codezahl der höheren Ziffer nach STRG.
        LD        A,(HL)       ;Abgriff der niederwertigen Ziffer
        AND       15           ;in der unteren Hälfte des Bytes
        ADD       48           ;und Addition von 48.
        LD        (STRG+1),A   ;Codezahl dieser Ziffer nach STRG+1.
        LD        HL,STRG      ;Textzeiger auf STRG,
        CALL      TEX          ;für das Programm TEX.
        POP       HL           ;Falls die Gesamtzahl aus mehr als zwei
        DEC       HL           ;Ziffern besteht, wird schon der
        INC       E            ;Zahlzeiger HL darauf vorbereitet
        INC       E            ;und auch DE korrigiert.
        BIT       7,D          ;Bei vergrößerter
        RET       Z            ;Darstellung springt
        INC       E            ;DE zwei weitere
        INC       E            ;Spalten nach rechts.
        RET
```

Routine zur Zifferndarstellung

```
        ENT       #5000        ;Der Objektcode wird oberhalb des unteren ROM
                               abgelegt
        ORG       #5000        ;und soll später an der gleichen Stelle aufgerufen
                               werden.
        CALL      #B906        ;Das untere ROM wird eingeblendet.
        LD        A,2          ;Der Bildschirmmodus 2
        CALL      SMO          ;wird eingeschaltet.
        LD        D,138        ;In vergrößerter Darstellung soll in Zeile 10
        LD        E,50         ;ab Spalte 50 ausgegeben werden.
        LD        HL,ZAHL+1    ;HL zeigt auf das obere Byte von ZAHL.
        CALL      ZIFF         ;Beide Byte von ZAHL
        CALL      ZIFF         ;werden ausgegeben.
        CALL      #B909        ;Das untere ROM wird wieder ausgeblendet.
        RET
SMO:    EQU       #0AE5        ;#0ACA beim CPC464, #0AE9 beim CPC6128
V:      DEFB      0,0,0        ;Datenblock für den Inhalt von STRG
ZAHL:   DEFB      137,21       ;Zahl, die ausgegeben werden soll, im BCD-Code
STRG:   EQU       V            ;Zeiger auf den Anfang des Datenblocks
```

Hauptprogramm für Schneider CPC zum Aufruf von ZIFF

```
        ORG       #            ;Ablage und Aufruf des Objektcodes hinter dem
                               BASIC
        LD        D,138        ;Ausgabe soll in vergrößerter Darstellung in
        LD        E,15         ;Zeile 10 ab Spalte 15 erfolgen.
        LD        HL,ZAHL+1    ;HL zeigt auf das obere Byte von ZAHL.
        CALL      ZIFF         ;Beide Byte von ZAHL
        CALL      ZIFF         ;werden ausgegeben.
        RET
V       DEFB      96,0,0,0     ;Datenblock für den Inhalt von IX+FARBE und STRG
ZAHL    DEFB      137,21       ;Zahl, die ausgegeben werden soll, im BCD-Code
STRG    EQU       V+1          ;Zeiger in den Datenblock
```

Hauptprogramm für Sinclair Spectrum zum Aufruf von ZIFF

ZX-ärgert mich

Ein lustiges Spiel für ein bis vier Personen, wobei der Spectrum (16 und 48K) bis zu vier Spieler übernehmen kann.

Die Spielregeln sind wie beim „Mensch ärgere Dich nicht"-Spiel, das heißt, am Anfang sind alle vier Figuren eines Spielers im „Gehäuse", das nur verlassen werden kann, wenn man eine Sechs würfelt. Es ist außerdem Pflicht, eine Figur ins Spiel zu bringen, falls man eine Sechs gewürfelt hat.

Ausnahmen:
– auf dem Raussetzfeld steht eine eigene Figur
– man hat alle vier Figuren im Spiel

Weiterhin ist es Pflicht, das Raussetzfeld freizumachen, ausgenommen man müßte seine eigene Figur schlagen. Wer eine sechs gewürfelt hat, darf nochmals würfeln. Dreimal würfeln darf man, wenn alle Figuren im „Gehäuse" stehen. Es besteht natürlich Zugzwang, auch wenn Sie nicht wollen. Ziel des Spieles ist es, alle vier Figuren in die „Einbahnstraße" zu lotsen.
Bedient wird das Spiel über die Tasten 1, 2, 3, 4, 0 und S. Die Tasten 1 bis 4 stehen für die einzelnen Figuren, die gezogen werden können. Wird die 0 gedrückt, so versteht der Computer das so, daß Sie nicht mehr ziehen können. Dies überprüft er natürlich. Sollte er nun eine Figur finden, die gezogen werden kann, so übernimmt der ZX diesen Zug. Sich um einen Zug drücken zu wollen kann also negative Folgen haben. Wenn Sie die Taste „S" drücken (und danach keine andere Taste mehr), dann beendet der ZX dieses Spiel. Viel Spaß bei dem Spiel „ZX ärgert mich".

Michael Silberberg

```
   0>REM  ZX aergert mich
          © maerz 1986 by
          Michael Silberberg
          Berliner Strasse 10
          5657 Haan
   5 CLEAR 49999
  10 DATA 33,0,64,17,80,195,1,0,
27,237,176,201
  20 RESTORE 10: FOR n=60000 TO
60011: READ a: POKE n,a: NEXT n
  30 POKE 50041,0
 100 RANDOMIZE : OVER 0: LET col
=1: BRIGHT 1: PAPER 7: INK 1: CL
S
 101 BRIGHT 0
 200 DIM p(4,4): DIM d(4): DIM w
(4): DIM q(4): DIM f(80): DIM s(
4,4)
 300 RESTORE 300: FOR n=1 TO 4:
READ a: FOR m=1 TO 4: LET p(n,m)
=m: LET f(a+m)=n: LET s(n,m)=a+m
: NEXT m: NEXT n
 310 DATA 56,60,64,68
 400 FOR n=1 TO 4: LET w(n)=4: N
EXT n
 450 LET gr=0: RANDOMIZE
 460 IF PEEK 50041=64 THEN GO TO
 550
 500 GO SUB 1000: GO SUB 1600: L
ET apl=0: FOR k=1 TO 4: RESTORE
500+k: READ col,pl: FOR l=0 TO 3
: LET nr=l+1: LET pl=pl+1: GO SU
B 1200: NEXT l: NEXT k
 501 DATA 0,56
 502 DATA 6,60
 503 DATA 4,64
 504 DATA 3,68
 505 RANDOMIZE USR 60000: GO TO
600
 550 POKE 60001,80: POKE 60002,1
95: POKE 60004,0: POKE 60005,64
 555 RANDOMIZE USR 60000
 600 FOR q=1 TO 4: LET qw=q: GO
SUB 1500: GO SUB 1700: PRINT AT
1,23;"Soll die";AT 2,23;"Farbe";
 INK col;"██": PRINT AT 3,23;"0)
 weg-";AT 4,23;"fallen";AT 5,23;
"1) von ";AT 6,23;"einem";AT 7,2
3;"Spieler";AT 8,23;",oder ";AT
9,23;"2) vom ZX"
 610 PRINT AT 10,23;"ueber-";AT
11,23;"nommen";AT 12,23;"werden
?"
 620 LET m$=INKEY$: IF CODE m$<4
8 OR CODE m$>50 THEN GO TO 620
 630 LET q(q)=VAL m$: NEXT q
 800 GO TO 5000
1000 REM Spielfeld
1010 RESTORE 1000: LET x=64: LET
 y=8
1020 FOR n=1 TO 12: READ a,b,c:
FOR m=1 TO c: LET x=x+16*a: LET
y=y+16*b: CIRCLE x,y,7: NEXT m:
NEXT n
1050 DATA 1,0,3,0,1,4,1,0,4,0,1,
2,-1,0,4,0,1,4,-1,0,2,0,-1,4,-1,
0,4,0,-1,2,1,0,4,0,-1,4
1060 DATA 96,24,0,1,0,32,88,1,0,
6,96,152,0,-1,4,160,88,-1,0,3
1070 FOR n=1 TO 4: READ x,y,a,b,
c
1080 INK c: FOR m=1 TO 4: CIRCLE
 x,y,4: LET x=x+16*a: LET y=y+16
*b: NEXT m: NEXT n
1090 INK 1: FOR n=1 TO 4: READ x
,y,a,b: PLOT x,y: DRAW 20*a,20*b
: DRAW -3+6*(a=-1),-3+6*(b=-1):
DRAW 3-6*(a=-1),3-6*(b=-1): DRAW
 3-6*(a=1),3-6*(b=1): NEXT n
1100 DATA 60,8,0,1,16,124,1,0,13
2,168,0,-1,172,52,-1,0
1110 RETURN
1200 REM Figurendruck
1210 OVER 1: LET co=(nr+48)*8+15
360: IF apl=0 THEN GO TO 1230
1215 IF apl>40 THEN INK col: GO
TO 1220
1216 INK 1
1220 RESTORE 1200: FOR n=1 TO ap
l: READ x,y: NEXT n: GO SUB 1250
1230 INK col: RESTORE 1200: FOR
n=1 TO pl: READ x,y: NEXT n: GO
SUB 1250
1240 INK 1: RETURN
1250 LET rg=0: LET c=0: FOR n=co
 TO co+7
1260 LET b=128: LET a=PEEK n: FO
R m=0 TO 7
1270 IF nr<>0 AND a>=b THEN GO S
UB 1301
1271 IF nr=0 AND a>=b THEN PLOT
x-c,y+m: LET a=a-b
1280 LET rg=rg+gr: LET b=b/2: NE
XT m
1290 LET rg=0: LET c=c+1+gr: NEX
T n
1300 OVER 0: RETURN
1301 FOR h=0 TO gr STEP 4: FOR j
```

```
=0 TO gr/2: PLOT x+m+rg+h,y-c-j:
 NEXT j: NEXT h: LET a=a-b: RETU
RN
1310 DATA 76,12,76,28,76,44,76,6
0,76,76,60,76,44,76,28,76,12,76,
12,92,12,108,28,108,44,108,60,10
8,76,108
1320 DATA 76,124,76,140,76,156,7
6,172,92,172,108,172,108,156,108
,140,108,124,108,108,124,108,140
,108,156,108,172,108
1330 DATA 172,92,172,76,156,76,1
40,76,124,76,108,76,108,60,108,4
4,108,28,108,12,92,12
1340 DATA 92,28,92,44,92,60,92,7
6,28,92,44,92,60,92,76,92,92,156
,92,140,92,124,92,108,156,92,140
,92,124,92,108,92
1350 DATA 12,12,28,12,12,28,28,2
8,12,172,28,172,12,156,28,156,15
6,172,172,172,156,156,172,156,15
6,28,172,28,156,12,172,12
1400 REM Wuerfel
1410 GO SUB 1520: LET wu=INT (RN
D*6+1): PLOT 221,6: DRAW 32,0: D
RAW 0,32: DRAW -32,0: DRAW 0,-32
1415 BEEP .01,1: BEEP .01,2: BEE
P .01,1
1420 IF wu=1 OR wu=5 OR wu=3 THE
N PRINT AT 19,29;"■"
1430 IF wu<>1 THEN PRINT AT 18,2
8;"■";AT 20,30;"■"
1440 IF wu=4 OR wu=5 OR wu=6 THE
N PRINT AT 20,28;"■";AT 18,30;"■
"
1450 IF wu=6 THEN PRINT AT 18,29
;"■";AT 20,29;"■"
1460 RETURN
1500 REM Loeschen
1510 OVER 0: FOR n=0 TO 21: PRIN
T AT n,23;"         ": NEXT n: R
ETURN
1520 OVER 0: FOR n=17 TO 21: PRI
NT AT n,23;"         ": NEXT n:
RETURN
1600 REM
1610 RESTORE 1600: LET x=0: LET
y=0: LET nr=0: FOR k=1 TO 22: RE
AD ms: LET m$=CHR$ ms: LET co=15
360+CODE m$*8: GO SUB 1250: LET
y=y+8: NEXT k
1620 DATA 32,90,88,32,97,101,114
,103,101,114,116,32,109,105,99,1
04,32,127,32,77,83,32
1630 OVER 1: FOR k=0 TO 21: PRIN
T AT k,0;" ": NEXT k: OVER 0: RE
TURN
1700 REM Farbe
1710 RESTORE 1710+qw
1711 DATA 0
1712 DATA 6
1713 DATA 4
1714 DATA 3
1720 READ col
1730 RETURN
1800 REM PRINT
1810 LET qw=q: GO SUB 1700: REST
ORE 1800: GO SUB 1500
1820 DATA "ist am","Zug"
1830 PRINT AT 1,23; INK col;"███
█████": FOR k=1 TO 2: READ m$:
PRINT AT 1+k,23;m$: NEXT k
1840 RETURN
4000 REM Spieler
```


Programm-Bibliothek

Wichtige Hilfsroutinen, auf die man immer wieder zurückgreifen kann (Teil 15)

28. Grafische Integration

Die nachfolgende Routine berechnet die Größe eines Flächenstückes, das zwischen einer Kurve und einer waagerechten Achse liegt. Aus der grafischen Darstellung wird, bei einer beliebigen unteren Grenze beginnend, eine ungerade Zahl von Hochwerten (y-Werte) in aufeinanderfolgenden, gleichweiten Schritten abgelesen. Je kleiner die Schrittweite gewählt wird, desto genauer ist die berechnete Fläche.

Der Aufruf

Dem Unterprogramm werden ab Zeile 1000 der Anfangswert U, die Schrittweite S, die ungerade Anzahl N und die Größe der abgelesenen Hochwerte Y (I) übergeben. Vor der Eingabe der Hochwerte erfolgt durch die Anlegung der für die Berechnung notwendigen Felder durch eine DIM-Anweisung (Zeile 160). Das Programm berechnet in der Variablen F die Größe des Flächenstückes in den entsprechenden Grenzen.

Die Routine

Die Berechnung der Größe der Fläche erfolgt über die sog. M-Formel der Simpson-Regel in Zeile 1070. Die dafür notwendigen Summen und Differenzen werden im Unterprogramm berechnet.

```
110 PRINT CHR$(147)
120 CLR
130 INPUT "UNTERE GRENZE";U
140 INPUT "SCHRITTWEITE";S
150 INPUT "ANZAHL DER WERTE";N
160 DIM D1(N),D2(N),Y(N)
210 OPEN 1,0
220 FOR I=0 TO N-1
230 X=U+I*S : PRINT :
240 PRINT I+1".WERT: "X" ";
250 INPUT#1,Y(I)
260 NEXT
270 CLOSE 1
500 GOSUB 1000
720 PRINT : PRINT
740 PRINT "FLAECHE:"F
760 PRINT
780 PRINT "TASTE !" : G$=""
800 GET G$ : IF G$="" THEN 800
820 PRINT
840 GOTO 120
1005 S1=0 : S2=0 : F=0 : N=N-1
1010 FOR I=1 TO N-1 STEP 2
1015 S1=S1+Y(I)
1020 NEXT
```

```
1025 FOR I=1 TO N
1030 D1(I)=Y(I)-Y(I-1)
1035 NEXT
1040 FOR I=1 TO N-1
1045 D2(I)=D1(I+1)-D1(I)
1050 NEXT
1055 FOR I=1 TO N-1 STEP 2
1060 S2=S2+D2(I)
1065 NEXT
1070 F=2*S*(S1+S2/6)
1075 RETURN
```

Die Routine

Das Intervall (U, O) wird in n Streifen der Breite S zerlegt (Zeile 1005). In Zeile 1020 erfolgt eine Korrektur der Streifenbreite S, wenn die Grenzen nicht ganzzahlig sind. Die Berechnung der Funktionswerte erfolgt in der Schleife ab Zeile 1020 bis Zeile 1030. Aus der Streifenbreite S der Zerlegung (siehe auch Zeile 1020) und den Summen S1, S2 erfolgt die Berechnung des Wertes für das Integral in Zeile 1095 in der Variablen F.

29. Bestimmtes Integral

Die nachfolgende Routine berechnet die Größe des bestimmten Integrals von beliebigen Funktionen mit einer Variablen.

Der Aufruf

Vor dem Sprung ins Unterprogramm und vor Übergabe der Werte muß eine CLR-Anweisung erfolgen. Dem Unterprogramm werden ab Zeile 1000 die obere sowie untere Grenze O, U und der Funktionsterm F(x), (z.B. ist F(x) = SIN (x) übergeben. Der berechnete Wert des Integrals wird in Zeile 770 ausgegeben.

```
110 PRINT CHR$(147)
120 CLR
130 DEF FNF(X)=SIN(X)
140 INPUT "OBERE GRENZE";O
150 INPUT "UNTERE GRENZE";U
500 GOSUB 1000
700 REM OUTPUT
760 PRINT
770 PRINT "FLAECHE: "F
780 PRINT
790 PRINT "TASTE !" : G$=""
800 GET G$ : IF G$="" THEN 800
820 PRINT
840 GOTO 120
1005 S=0.02 : N=INT((O-U)/S+0.1)
1010 IF N/2=INT(N/2) THEN N=N+1
1015 DIM D1(N),D2(N),Y(N),X(N)
1020 S=(O-U)/(N-1) : FOR I=0 TO N
1025 X(I)=U+I*S : Y(I)=FNF(X(I))
1030 NEXT
1035 S1=0 : S2=0 - F=0 : N=N-1
1040 FOR I=1 TO N-1 STEP 2
1045 S1=S1+Y(I) : NEXT
1050 FOR I=1 TO N
1055 D1(I)=Y(I)-Y(I-1)
1060 NEXT
1065 FOR I=1 TO N-1
1070 D2(I)=D1(I+1)-D1(I)
1075 NEXT
1080 FOR I=1 TO N-1 STEP 2
1085 S2=S2+D2(I)
1090 NEXT
1095 F=2*S*(S1+S2/6)
1100 RETURN
```

```
4005 GO SUB 9100: LET re=wq
4010 GO SUB 1500: GO SUB 1800
4020 IF wq=0 THEN RETURN
4030 GO SUB 1400: LET wq=wq-1
4035 IF re=3 AND wu<>6 THEN PAUS
E 5: PAUSE 100: GO TO 4020
4100 DATA "Mit","welcher","Figur
","wollen","Sie","ziehen ?"
4110 RESTORE 4100: FOR k=1 TO 6:
 READ m$: PRINT AT 5+k,23;m$: NE
XT k
4115 LET wq=0+(wu=6): LET re=0
4120 LET m$=INKEY$: IF CODE m$<4
8 OR CODE m$>52 THEN GO TO 4120
4125 IF m$="0" THEN GO TO 6026
4130 BEEP .05,2: LET a=VAL m$: I
F s(q,a)>56 AND wu<>6 THEN GO TO
 4900
4140 IF f(q*10-9)=q AND f(q*10-9
+wu)<>q AND s(q,a)<>q*10-9 THEN
GO TO 4900
4145 RESTORE 6170: FOR k=1 TO q:
 READ c: NEXT k
4151 IF f(s(q,a)+wu)=q AND (s(q,
a))<=40 AND (s(q,a))+wu>40 AND c
=40 THEN GO TO 4900
4152 IF f(s(q,a)+wu)=q AND s(q,a
)+wu<=40 AND (((s(q,a))<c AND (s
(q,a))+wu<c) OR ((s(q,a))>c AND
(s(q,a))+wu>c)) THEN GO TO 4900
4160 LET pl=0: LET b=0: FOR k=1
TO 4: LET b=b+(f(q*4+k+52)=q): N
EXT k: IF wu=6 AND f(q*10-9)<>q
AND s(q,a)<57 AND b<>0 THEN GO T
O 4900
4170 IF s(q,a)>56 THEN LET pl=q*
10-9
4175 IF s(q,a)+wu>40 AND s(q,a)<
=40 AND q<>1 THEN LET pl=s(q,a)+
wu-40
4176 IF pl<>0 THEN GO TO 4200
4185 IF c=40 AND s(q,a)+wu<=44 T
HEN LET pl=s(q,a)+wu: GO TO 4200
4186 IF s(q,a)<=c AND s(q,a)+wu>
c THEN LET wu1=s(q,a)-c+wu: LET
pl=4*q+36+wu1: GO TO 4200
4190 IF s(q,a)>=4*q+37 AND s(q,a
)+wu<=4*q+40 THEN LET pl=s(q,a)+
wu: GO TO 4200
4195 IF s(q,a)+wu<=40 THEN LET p
l=s(q,a)+wu
4196 IF pl=0 THEN GO TO 4900
4200 IF f(pl)=q AND pl>40 OR pl>
q*4+40 THEN GO TO 4900
4201 LET nr=a: LET nr1=nr: LET a
pl=s(q,nr): IF pl>40 AND apl<=40
 THEN LET d(q)=d(q)+1
4202 IF apl>56 THEN LET w(q)=w(q
)-1
4203 GO SUB 8000: LET qw=q: LET
nr=nr1: GO SUB 1700: GO SUB 1200
: LET s(q,nr1)=pl: LET f(apl)=0:
 LET f(pl)=q: GO TO 4020
4900 PRINT AT 13,23;"Illegal";AT
 14,23;"move !": PAUSE 30: BEEP
.1,7: BEEP .1,5: BEEP .1,7: PAUS
E 111: PRINT AT 13,23;"        ";
AT 14,23;"        ": GO TO 4110
5000 REM Start
5010 PAPER 7: INK 1: GO SUB 1500
5500 FOR q=1 TO 4
5505 IF d(q)=4 THEN GO TO 5530
5510 IF q(q)=1 THEN GO SUB 4000
5520 IF q(q)=2 THEN GO SUB 6000
```

```
5525 IF PEEK 23560=CODE "s" OR P
EEK 23560=CODE "S" THEN GO TO 95
00
5530 NEXT q
5600 LET zx=0: FOR q=1 TO 4: LET
 zx=zx+(d(q)=4 OR q(q)=0): NEXT
q: IF zx>=3 THEN GO TO 9500
5999 GO TO 5000
6000 REM zz
6010 GO SUB 1700: GO SUB 1800: G
O SUB 9000
6020 IF wq=0 THEN RETURN
6021 IF wq=3 THEN FOR k=1 TO 4:
LET p(q,k)=k: NEXT k
6025 GO SUB 1400: LET wq=wq-1
6026 IF wu=6 THEN LET wq=1
6027 IF f(q*10-9)=q AND f(q*10-9
+wu)<>q THEN GO TO 8900
6028 IF wu+d(q)<=4 AND d(q)>0 TH
EN GO TO 7000
6030 IF wu<>6 THEN GO TO 6100
6050 IF w(q)=0 THEN GO TO 6100
6060 IF f(q*10-9)=q THEN GO TO 6
100
6070 LET w(q)=w(q)-1: GO SUB 880
0: LET apl=q*4+52+bn: LET pl=q*1
0-9: GO SUB 8000
6080 LET qw=q: LET f(apl)=0: LET
 f(pl)=q: FOR k=1 TO 4: IF s(q,k
)=apl THEN LET zx=k
6090 NEXT k: LET s(q,zx)=pl: LET
 nr=zx: GO SUB 1700: GO SUB 1200
: GO TO 6020
6100 LET aq=wq: GO SUB 9000: IF
wq=3 THEN LET wq=aq: PAUSE 5: PA
USE 100: GO TO 6020
6110 LET wq=0+(wu=6): GO TO 7100
6120 FOR k=1 TO 4: LET a1=s(q,k)
+wu: IF a1-wu>40 THEN GO TO 6150
6125 IF a1>40 THEN LET a1=a1-40
6130 IF f(a1)<>0 AND f(a1)<>q TH
EN GO TO 6160
6150 NEXT k: GO TO 6200
6160 RESTORE 6160: FOR l=1 TO q:
 READ b1: NEXT l
6170 DATA 40,10,20,30
6180 IF a1>b1 AND b1>=s(q,k) THE
N GO TO 6150
6185 IF b1=40 AND a1-wu<1 THEN G
O TO 6150
6190 LET nr=k: LET apl=s(q,k): L
ET pl=a1: GO SUB 8000: LET nr=k:
 LET qw=q: GO SUB 1700: GO SUB 1
200: LET s(q,k)=pl: LET f(apl)=0
: LET f(pl)=q: GO TO 6020
6200 FOR z=1 TO 4: LET p=p(q,z):
 IF s(q,p)>40 THEN GO TO 6300
6210 LET b1=wu+s(q,p): RESTORE 6
170: FOR k=1 TO q: READ c1: NEXT
 k
6220 IF b1>40 THEN LET b1=b1-40
6230 IF c1=40 AND b1-wu<1 THEN G
O TO 6300
6240 IF b1>c1 AND c1>=s(q,p) THE
N GO TO 6300
6245 IF f(b1)=q THEN GO TO 6300
6250 LET nr=p: LET apl=s(q,nr):
LET pl=b1: GO SUB 8000: LET qw=q
: GO SUB 1700: GO SUB 1200: LET
s(q,nr)=pl: LET f(apl)=0: LET f(
pl)=q: GO TO 6020
6300 NEXT z
6950 IF q(q)=1 THEN BEEP .2,-9:
GO TO 4020
6955 RESTORE 6950: FOR k=1 TO 3:
 READ m$: PRINT AT 5+k,23;m$: NE
XT k: BEEP .1,5: PAUSE 5: PAUSE
100
6960 DATA "Ich kann","nicht","zi
ehen !"
6999 GO TO 6020
7000 FOR k=1 TO 4: IF f(36+4*q+k
)=q AND 36+4*q+k+wu<=40+4*q THEN
 LET apl=36+4*q+k: LET pl=36+4*q
+k+wu: GO TO 7020
7010 NEXT k: GO TO 6029
7020 IF f(pl)=q THEN GO TO 6029
7030 FOR k=1 TO 4: IF s(q,k)=apl
 THEN LET nr=k
7040 NEXT k
7050 LET qw=q: GO SUB 1700: GO S
UB 1200: LET s(q,nr)=pl: LET f(a
pl)=0: LET f(pl)=q: GO TO 6020
7100 LET nr=0: RESTORE 6170: FOR
 k=1 TO q: READ a: NEXT k
7110 LET b=a-wu: FOR k=1 TO 4: I
F s(q,k)>=b AND s(q,k)<=a THEN L
ET nr=k
7120 NEXT k: IF nr=0 THEN GO TO
6120
7123 LET b=s(q,nr)+wu: IF b>a TH
EN LET b=(b-a)+36+q*4
7124 IF b>4*q+40 THEN GO TO 6120
7125 IF b<=40 THEN GO TO 6120
7130 IF f(b)=q THEN GO TO 6120
7140 LET d(q)=d(q)+1: LET apl=s(
q,nr): LET pl=b: LET nr1=nr: GO
SUB 8000: LET qw=q: LET nr=nr1:
GO SUB 1700: GO SUB 1200: LET s(
q,nr)=pl: LET f(apl)=0: LET f(pl
)=q: GO TO 6020
8000 REM Raus
8010 IF f(pl)=0 THEN RETURN
8020 LET ms=f(pl): LET w(ms)=w(m
s)+1: LET f(pl)=0: FOR l=1 TO 4:
 IF s(ms,l)=pl THEN LET zx=l
8030 NEXT l: LET xz=52+ms*4+zx:
LET f(xz)=ms: LET s(ms,zx)=xz: L
ET apl1=apl: LET pl1=pl: LET apl
=pl: LET nr=zx: LET pl=xz: LET q
w=ms: GO SUB 1700: GO SUB 1200
8035 GO SUB 8100
8040 LET apl=apl1: LET pl=pl1: R
ETURN
8100 FOR o=1 TO 4: IF p(ms,o)=nr
 THEN LET zx=o
8110 NEXT o: FOR o=zx TO 3: LET
p(ms,o)=p(ms,o+1): NEXT o: LET p
(ms,4)=nr: NEXT o: RETURN
8800 REM apl
8810 FOR l=1 TO 4: IF f(q*4+52+l
)=q THEN LET bn=l: RETURN
8820 NEXT l: LET bn=4: RETURN
8900 FOR k=1 TO 4: IF s(q,k)=q*1
0-9 THEN LET nr1=k
8910 NEXT k
8920 LET apl=q*10-9: LET pl=q*10
-9+wu: GO SUB 8000: LET nr=nr1:
LET qw=q: GO SUB 1700: GO SUB 12
00: LET s(q,nr1)=pl: LET f(apl)=
0: LET f(pl)=q: GO TO 6020
9000 REM Pruef
9020 LET a=q*4+40: IF w(q)=4 THE
N GO TO 9090
9030 IF w(q)=3 AND f(a)=q THEN G
O TO 9090
9040 IF w(q)=2 AND f(a)=q AND f(
a-1)=q THEN GO TO 9090
```

```
9050 IF w(q)=1 AND f(a)=q AND f(
a-1)=q AND f(a-2)=q THEN GO TO 9
090
9060 LET wq=1: GO TO 9095
9090 LET wq=3
9095 RETURN
9100 REM Pruef 5
9110 LET wq=0: FOR k=1 TO 4: IF
f(q*4+52+k)=q THEN LET wq=wq+1
9115 NEXT k
9120 IF wq=4 THEN LET wq=3: RETU
RN
9130 IF wq=3 AND f(q*4+40)=q THE
N RETURN
9140 IF wq=2 AND f(q*4+40)=q AND
 f(q*4+39)=q THEN LET wq=3: RETU
RN
9150 IF wq=1 AND f(q*4+40)=q AND
 f(q*4+39)=q AND f(q*4+38)=q THE
N LET wq=3: RETURN
9160 LET wq=1: RETURN
9500 PAUSE 222: PAPER 7: BRIGHT
1: CLS : GO SUB 9800: RESTORE 95
50: PRINT AT 18,0;"Druecken Sie
'Y' um nochmal zu  spielen !"
9550 READ ms,zx: IF ms=18 THEN R
ESTORE 9550: PAUSE 5: GO TO 9550
9555 BEEP .4/ms,zx
9556 DATA 4,8,4,8,4,6,4,8,2,-1,4
;1,4,4,8,4,8,6,2,6,4,-59,4,8,4,8
,8,6,8,8,4,6,2,4,4,-59
9557 DATA 4,6,4,6,8,4,8,6,4,4,2,
3,4,-59,4,8,4,8,8,6,8,8,4,6,8,4,
16,8,8,8,16,6,4,8,2,-1,2,1,4,4,8
,4,8,6
9558 DATA 2,6,4,-59,4,8,4,8,8,6,
8,8,4,6,4,4,4,4,8,6,8,6,8,6,8,6,
4,6,4;9,2,8,2,6,1,4,18,18
9600 LET m$=INKEY$: IF m$<>"y" A
ND m$<>"Y" THEN GO TO 9550
9700 GO TO 100
9800 RESTORE 9800: LET gr=4: LET
 nr=1: FOR k=1 TO 13: INK INT (R
ND*4): READ m$,x,y: LET co=15360
+CODE m$*8: GO SUB 1250: NEXT k
9810 DATA "Z",48,170,"X",176,170
,"a",4,130,"e",40,130,"r",76,130
,"g",112,130,"e",148,130,"r",184
,130,"t",220,130,"m",16,90,"i",8
0,90,"c",144,90,"h",208,90
9820 PRINT #0;" © 1986 by Michae
l Silberberg"
9830 RETURN
```

UDG+

Dieses Programm erweitert und ergänzt die Blockgrafik des Spectrum und macht es möglich, ansehnliche Grafikfiguren zu entwerfen, ohne spezielle UDG-Zeichen zu konstruieren. Insgesamt stehen 72 Standard-Grafikzeichen zur Verfügung (Spectrum 48K).

In Zeile 9910 wird der normale Zeichensatz aus dem ROM in die Nähe des RAM-Tops verschoben. Die Zeile 9920 legt zunächst die Start-Adresse des neuen Zeichensatzes fest (POKE 23607,249), wonach in der folgenden FOR-NEXT-Schleife die neuen Zeichen eingelesen und in die neuen Adressen gepoket werden. Zur Übersicht werden innerhalb der Zeilen 9992 bis 9996 durch eine Schleife die betreffenden Zeichen mittels CHR§ normal und invers dargestellt.

Bedienungshinweise

Die Bedienung von ,,UDG+" ist denkbar einfach:

1. ,,UDG+" durch LOAD"" laden oder durch MERGE"" dazuladen. (Vor MERGE"" ist es ratsam, sich zu vergewissern, ob die Zeilen 9900 bis 9997 frei sind).
2. Durch GOSUB 9900 werden alle Kleinbuchstaben nach kurzer Wartezeit zu Grafikzeichen umgewandelt. (Bitte beachten Sie, daß dann alle Kleinbuchstaben nicht mehr verfügbar sind. Das gilt auch für Fehlermeldungen.)
3. Mit GOTO 9990 werden die betreffenden Zeichen in einer Tabelle in der Reihenfolge Buchstabe, Grafikzeichen normal und Grafikzeichen invers übersichtlich gemacht. (Ein Ausdrucken dieser Tabelle mit COPY erleichtert das Arbeiten erheblich.)

- Sie können ,,UDG+" auch dazu benutzen, die DATA-Zeilen mit eigenen Entwürfen zu füllen.
- Sollten Sie ,,UDG+" in Verbindung mit einem anderen Programm benutzen, müssen Großbuchstaben für die Variablen genommen werden, damit das Listing lesbar bleibt.

Peter Bergen

```
9900 REM UDG+
9910 CLEAR 63999: LET X=48384: F
OR Y=15616 TO 16383: POKE (Y+X),
PEEK Y: NEXT Y
9920 POKE 23607,249: RESTORE 902
0: FOR Y=1 TO 26: READ A$: LET Z
=63744+8*CODE A$: FOR T=0 TO 7:
READ S: POKE Z+T,S: NEXT T: NEXT
 Y: RETURN
9930 DATA "a",0,0,0,0,1,3,7,15
9932 DATA "b",0,0,0,0,128,192,22
4,240
9934 DATA "c",240,224,192,128,0,
0,0,0
9936 DATA "d",15,7,3,1,0,0,0,0
9938 DATA "e",0,0,0,0,24,60,126,
255
9940 DATA "f",128,192,224,240,24
0,224,192,128
9942 DATA "g",255,126,60,24,0,0,
0,0
9944 DATA "h",1,3,7,15,15,7,3,1
9946 DATA "i",1,1,3,3,7,15,63,25
5
9948 DATA "j",128,128,192,192,22
4,240,252,252
9950 DATA "k",255,252,240,224,19
2,192,128,128
9952 DATA "l",255,63,15,7,3,3,1,
1
9954 DATA "m",1,3,7,15,31,63,127
,255
9956 DATA "n",128,192,224,240,24
8,252,254,255
9958 DATA "o",255,231,195,129,12
9,195,231,255
9960 DATA "p",255,126,60,24,24,6
0,126,255
9962 DATA "q",153,204,102,51,153
```

```
,204,102,51
9964 DATA "r",255,129,255,24,255
,129,255,24
9966 DATA "s",153,102,102,153,15
3,102,102,153
9968 DATA "t",85,170,85,170,85,1
70,85,170
9970 DATA "u",85,0,85,0,85,0,85,
0
9972 DATA "v",195,129,0,0,0,0,12
9,195
9974 DATA "w",24,60,126,255,255,
126,60,24
9976 DATA "x",24,60,126,255,255,
126,24,24
9978 DATA "y",102,255,255,255,12
6,60,24,0
9980 DATA "z",24,60,24,90,255,25
5,90,24
9990 REM UDG+DEMO
9992 PAPER 7: INK 0: CLS : LET V
E=1: LET HO=0: FOR Y=65 TO 90: P
RINT AT VE,HO;CHR$ (Y);" = ";CHR
$ (Y+32);"  "; INVERSE 1;CHR$ (Y
+32): LET VE=VE+2
9994 IF VE>20 THEN LET VE=1: LET
 HO=HO+11
9996 NEXT Y: STOP
```

Variable	Bedeutung
HO	Horizontale für Tabelle
S	Variable zur Aufnahme der DATAS
T	FOR-NEXT-Variable zum Einlesen der DATAS
VE	Vertikale für Tabelle
X	Basis der Zeichensatzverschiebung
Y	Universelle FOR-NEXT-Variable
Z	Adressen der neuen Zeichen
A§	String zur Aufnahme der Kleinbuchstaben

Variablenliste

A =	K =	U =
B =	L =	V =
C =	M =	W =
D =	N =	X =
E =	O =	Y =
F =	P =	Z =
G =	Q =	
H =	R =	
I =	S =	
J =	T =	

Tabelle der Grafikzeichen

Spickzettel

Die aktuelle Kurzanleitung

„DB Master One"
Dateiverwaltung für Atari ST

Kapazität:
Maximale Dateigröße: 320 KByte
Maximale Datensatzlänge: 3000 Zeichen
Maximale Feldlänge: 3000 Zeichen (Feldname höchstens 64 Zeichen)

MAKEONE

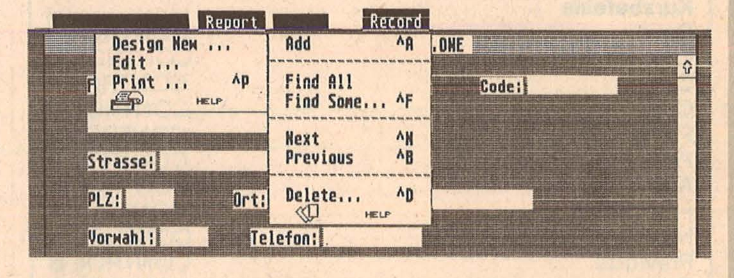
Um in den Programmteil „Dateimasken erstellen" zu gelangen, auf dem Desktop MAKEONE.PRG anklicken.

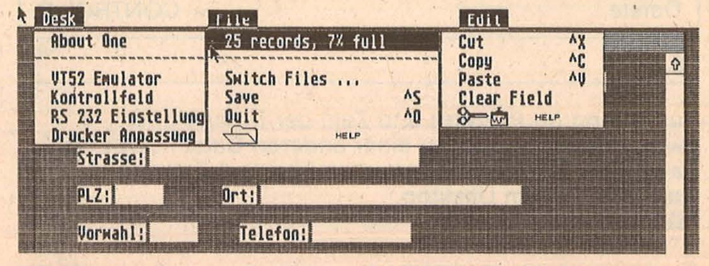
FILE

New: Anklicken, um eine neue Dateimaske zu erstellen.
Redesign: Abändern einer bestehenden Maske ohne Datenverlust.
Save: Abspeichern der aktuellen Maske.
Quit: Zurück zum Desktop.
Anklicken eines Punktes auf dem Arbeitsblatt positioniert das erste Datenfeld. Nach Eingabe des Feldnamens und RETURN erscheint die Kapazitätsangabe des Datenfeldes. Anklicken an der linken Punktmarkierung gestattet horizontale und vertikale Veränderung der Feldgrenzen (und der Kapazität). Nach dem Erstellen der Namen- und Datenfelder mit OPTIONS Schrifttyp und -größe festlegen. Absaven.

EDIT

Cut Field: Speichert ein ganzes Feld ab bis zur späteren Verwendung. (Das Feld verschwindet vom Arbeitsblatt.)
Copy Field: Speichert ein Feld ab. Kann mit „Paste" wieder aufgerufen werden.
Paste Field: Ersetzt ein aktuelles markiertes Feld durch dasjenige, das zuletzt mit „Copy" oder „Cut" abgespeichert wurde.
Cut Text: Speichert einen markierten Textteil zur späteren Verwendung ab.
Copy Text: Speichert einen markierten Textteil zur späteren Verwendung ab. Der Textteil bleibt auf dem Monitor.
Paste Text: Holt den zuletzt mit „Cut" oder „Paste" abgelegten Textteil an die Cursorposition zurück. Kurzbefehle entsprechend wie in USEONE





DB Master One — die Submenüs USEONE: Die Kurzbefehle gelten teilweise auch für MAKEONE

Speed-Dos neu

Nützliche Änderungen an einem Speed-Dos-System werden hier beschrieben. Es handelt sich um Anpassungen der Farben und Veränderungen der Betriebsweise des Monitors. Die neue Version kann entweder nach jedem Einschalten im RAM installiert werden oder, was bequemer ist, in ein EPROM gebrannt werden.

Bei den meisten Varianten von Speed-dos wird die Farbe des Bildschirms in den Speicherzellen ECD9 und EDCA durch 00 auf Schwarz eingestellt. Auf **schwarzem Bildschirm** sind Zeichen sehr kontrastreich, wenn sie **gelb** geschrieben werden. Dies wird durch folgende Eingriffe erreicht:
E4A9 = 9E bewirkt gelbe Zeichendarstellung **nach Startmeldung.**
E534 = lda #07 erzeugt gelbe Zeichenfarbe beim Bildschirmreset.
FF22 = 9E stellt das ,,Directory" (F1-Taste) in gelben Zeichen dar.

Auto-Repeat
Der **Auto-Repeat** für alle Tasten läßt sich leicht durch eine kurze Routine erreichen. Platz für sie ist ab EA61.
Die **Reset-Routine** ab FCE2 wird durch FCFB = JSR EA65 verlassen.
EA61 = JMP EA7B überspringt die ab EA65 eingefügte Routine.
Ab EA65 wird dann

```
JSR FF5B
LDA #80
STA 028A
RTS
```

eingefügt, die Auto-Repeat-Routine ist damit implementiert. Geänderte Tastenbelegung für Suchen und Ändern im Monitor: F1 erhöht momentane Adresse um 8, F3 verringert entsprechend, F5 erhöht um CO (entspricht einer Monitorseite), F7 verringert um diesen Wert.
Diese Änderungen werden in der Tabelle FB97 bzw. FBEO vorgenommen:

```
fb97   07 06 04 02 03 05
fbe0   01 c0 08
```

Tabelle mit neuen Werten
Für die beabsichtigten Änderungen sind sehr massive Eingriffe notwendig.
Die neue Anzeigenroutine des Monitors ist in den abgedruckten Programmen mit einem jeweils angefügten Kommentar beschrieben.

Funktionstasten-belegung
Es können die Belegungen für die Tasten F4, F6 und F8 geändert werden. Mit F4 wird jetzt (List), mit F6 sY 49152 (sys 49152) und mit F8 sA (save) erzeugt. Diese Änderungen werden durch Eingriff in die **Tabelle ab FF10** bewirkt.

```
ff1d   00 S3 90 3e 24 9e 0d 00   ...>$...   ;chr$(147),sw,dir.,
                                             ge,cr
ff25   S3 52 d5 0d 00 4c cf 22   .rU..l0"   ;chr$(147),run,load",
ff2d   00 4c cf 20 0d 00 00 4c   .l0...l    ; load#*#,cr,list
ff35   c9 0d 0d 00 53 d9 20 34   l...sY4    ;cr,cr,sys 49152,
ff3d   39 31 35 32 0d 00 53 c1   9152..sA   ; cr,save"
ff45   22 00
```

Geänderte Funktionstastenbelegung

SPLAT: Zugriff auf vorgefertigte Masken.
Mailing List: Adressenverzeichnis (Amerikanische Norm).
Checkbook: Schecknachweis.
Collection: Einfache Lager-/Rechnungsverwaltung.
Date: Jahr, Monat, Tagesdatum.
Wart Hogs: Hundezüchterkartei (Stubenrein, bissig?).

OPTIONS

Label Style: Schriftart und -größe des Feldnamens.
Data Style: Schriftart und -größe im Datenfeld.
Fünf Schriftgrößen wählbar. Außerdem fett, hell, kursiv, unterstreichen und Kontur.

USEONE

Mit QUIT zum Desktop und USEONE.PRG anklicken. Auf Verzeichnis der Masken Auswahl treffen, dann OK.

Kurzbefehle	
Save	CONTROL S
Quit	CONTROL Q
Cut	CONTROL X
Copy	CONTROL C
Paste	CONTROL V
Print	CONTROL P
Add	CONTROL A
Find Some	CONTROL F
Next	CONTROL N
Previous	CONTROL B
Delete	CONTROL D

FILE

Auslastung (in Prozent) und Zahl der Datenblätter.
Switch Files: Wechsel zu einer anderen Datei in USEONE.
Save: Abspeichern des aktuellen Files auf Diskette.
Quit: Zurück zum Desktop.

EDIT

Cut: Entfernt markierten Text vom Bildschirm in den Speicher. Er wird mit **Paste** zurückgerufen.
Copy: Markierter Text wird abgespeichert. Wird mit ,,Paste" wieder abgerufen.
Paste: Ruft den letzten mit **Copy** oder **Cut** abgelegten Text an die Cursorposition zurück.
Clear: Löscht alle Daten eines Feldes.

REPORT

Design New: Aufruf zu Sortieren und Ausgabe der Daten. Am unteren Bildrand erscheint ein Submenü mit den Optionen **Form, Fields, Find, Sort.**
Form: Column – Die Daten eines Blattes in einer Zeile.
Page – Die Daten eines Blattes als Briefkopf.
Record: Kompletter Ausdruck eines Datenblattes.
Fields: Bei den Optionen **Column** und **Page** Wahl der Felder, die ausgegeben werden sollen.
Find: Sucht exakt gleiche Begriffe in bestimmten Feldern (EQUALS), Teile von Zeichenketten (CONTAINS) oder Datenmengen innerhalb definierter Grenzen (BETWEEN), also ,,von – bis". Drei Suchkombinationen möglich.
Sort: Sortiert auf maximal drei Feldern gleichzeitig nach Alphabet oder Nummern, aufsteigend oder absteigend. Addiert Zwischensummen (Print Subtotals). Befehl für Seitenwechsel (Start New Page).
Edit: Mit der Löschfunktion (DELETE) können mit **Design New** erstellte Dateien aus dem ,,Report-Verzeichnis" gelöscht werden.
Print: Aufforderung, die auszudruckende Kartei zu benennen oder eine bereits vorhandene umzubenennen.
Aufforderung, die gewünschte Datei aus dem Report-Verzeichnis zu wählen und auf dem Monitor (Screen) zu zeigen, auszudrucken (Printer) oder auf Diskette abzulegen..

RECORD

Add: Neues Datenblatt derselben Kartei.
Find all: Führt zum Anfang einer Kartei.
Find Some: Sucht gleiche Begriffe in bestimmten Feldern. (EQUALS), Teile von Zeichenketten (CONTAINS) oder Datenmengen innerhalb definierter Grenzen (BETWEEN).
Next: Blättert zum nächsten Datenblatt.
Previous: Blättert zum vorhergehenden Datenblatt.
Delete: Löscht das aktuelle Datenblatt.



Monitordarstellung
Der alte Monitor hat eine gepackte Darstellung der Hex-Zahlen mit wechselnder reverser Darstellung, was wohl die Übersicht verbessern sollte. Die Zeichendarstellung der Hexzahlen im unteren Bildbereich macht die Zuordnung zwischen Zeichen und Hexzahl schwierig. Die Darstellung kann deshalb folgendermaßen geändert werden:
Nur acht Hexzahlen statt 16 pro Zeile
24 Zeilen mit acht Zeichen
Ein Blankzeichen zwischen den Zahlen
Abwechselnde reverse Darstellung entfällt
Zeichendarstellung der acht Hexzahlen **neben** den Zahlen mit folgender Vereinbarung: Zeichen mit CHR$-Code <20 werden als „." dargestellt, Zeichen mit CHR$-Code >127 als reverses „."
Als letzte Änderung muß in FA58 und FA5A die reverse Darstellung durch NOP aufgehoben werden.

Implementierung des neuen Systems
Die Änderungen werden von einem Monitor aus von E400 bis FD00 auf Diskette zurückgeschrieben. Wer nicht in der Lage ist, sich ein neues EPROM mit den Änderungen zu brennen, kann das neue System softwaremäßig nutzen. Dazu wird zuerst UP5 geladen und gestartet. Danach wird mit POKE 1,53 auf das ins RAM verschobene System umgeschaltet. Schließlich wird dann die Änderung geladen, und das neue System steht zur Verfügung. Beim erneuten Einschalten muß die Prozedur wiederholt werden. Deshalb ist ein neues EPROM auf Dauer vorteilhafter.

```
fa6c  a9 29     lda #29       ;Zeiger auf 0429 stellen
fa6e  85 fc     sta fc
fa70  ad 88 02  lda 0288      ;HByte von Bildspeicher
fa73  85 fd     sta fd        ;nach fd
fa75  a2 18     ldx #18       ;24 Zeilen schreiben
fa77  a9 00     lda #00       ;Spaltenzähler
fa79  85 c3     sta c3        ;löschen
fa7b  20 3d fa  jsr fa3d      ;Adresse anzeigen
fa7e  30 60 fa  jsr fa60      ;2 Leerzeichen
fa81  a4 c3     ldy c3        ;offset laden
fa83  b1 fe     lda (fe), Y   ;Inhalt laden
fa85  20 44 fa  jsr fa44      ;und anzeigen
fa88  e6 c3     inc c3        ;nächste Stelle
fa8a  20 5d fa  jsr fa5d      ;1 Leerzeichen
fa8d  a0 07     ldy #07       ;fertig?
fa8f  c4 c3     cpy c3
fa91  10 EE     bpl fa81      ;Schleife
fa93  20 5d fa  jsr fa5d      ;1 Leerzeichen
fa96  20 a2 fa  jsr faa2      ;Bildschirmcode
fa99  ca        dex           ;nächste Zeile
fa9a  d0 db     bne fa77      ;Schleife für 24 Zeilen
fa9c  20 ca fc  jsr fcca      ;Adressenzeiger wieder
fa9f  c6 ff     dec ff        ;herstellen
faa1  60        rts           ;Ende
```

Geänderte Anzeigeroutine des Monitors

```
faa2  b1 fe     lda (fe),y    ;Inhalt laden
faa4  20 5f f6  jsr f65f      ;Routine wandelt CHR$-
                              Code in
                              ;Bildschirmcode. Code <20
                              werden
                              ;als „.",
                              ;>127 als RVS „." dargestellt.
faa7  88        dey           ;nächstes Zeichen
faa8  10 f8     bpl faa2      ;darstellen
faaa  20 6e ea  jsr ea6e      ;nächste Zeile vorbereiten
faad  c8        iny           ;y löschen, weil nachfolgen-
                              des
                              ;Programm dies verlangt
faae  60        rts
```

UP1 wandelt Hexzahlen in Bildschirmcode um

```
f65f  c9 20     cmp #20       ;kleiner 20?
f661  90 11     bcc f674      ;ja
f663  c9 40     cmp #40       ;kleiner 40 -Ziffern-?
f665  90 0a     bcc f671      ;ja
f667  c9 60     cmp #60       ;kleiner 60 -Buchstaben-?
f669  90 0d     bcc f678      ;ja
f66b  c9 80     cmp #80       ;größer 127 -kein Bildcode-?
f66d  10 0d     bpl f67c      ;ja
f66f  49 20     eor #20       ;Code korrigieren/Grafikzeichen
f671  91 fc     sta (fc),y    ;darstellen
f673  60        rts           ;ende des up
f674  a9 2e     lda #2e       ;Code für „."
f676  10 f9     bpl f6771     ;„." darstellen
f678  49 40     eor #40       ;Codekorrigieren-Buchstaben-
f67a  10 f5     bpl f671      ;und darstellen
f67c  a9 ae     lda #ae       ;Code für RVS „." laden
f67e  30 f1     bmi f671      ;und darstellen
```

UP2 wandelt CHR$-Code in Bildschirmcode um

```
fcca  a9 40     lda #40       ;Komplement auf 100
fccc  20 22 fa  jsr fa22      ;alter Zeiger hergestellt
fccf  60        rts           ;durch dec ff außerhalb
```

UP3 stellt Bildzeiger auf Bildanfang

```
ea6e  a9 08     lda #08       ;offset laden
ea70  20 22 fa  jsr fa22      ;Adreßzeiger erhöhen
ea73  a9 09     lda #09       ;Zeilenoffset
ea75  20 62 fa  jsr fa62      ;Bildschirmzeiger auf
ea78  60        rts           ;Zeilenanfang stellen
```

UP4 erhöht Adreßzeiger um 8, der Bildschirmzeiger wird auf Zeilenanfang gestellt

```
033c  a9 00     lda #00       ;
033e  85 fe     sta fe        ;
0340  a2 01     ldx #01       ;Zeiger auf Tabelle
0342  bd 5d 03  lda 035 d,x
0345  85 ff     sta ff        ;Zeiger laden
0347  a0 00     ldy #00
0349  b1 fe     lda (fe),y    ;ROM laden
034b  91 fe     sta (fe),y    ;nach RAM
034d  88        dey           ;
034e  d0 f9     bne 0349      ;Schleife
0350  e6 ff     inc ff
0352  a5 ff     lda ff        ;fertig?
0354  dd 5f 03  cmp 035f,x
0357  d0 ee     bne 0342      ;nein
0359  ca        dex           ;nächstes ROM
035a  10 e6     bpl 0342
035c  60        rts           ;ende
035d  a0 e0 c0
```

UP5 verschiebt Betriebssystem ins RAM

SCROLL und CLEAR

Ein beliebiger Teil des Bildschirms kann vom Scrolling ausgeschlossen werden. Ferner eine Routine zum teilweisen Löschen des Bildschirms (C64).

Jeder Programmierer eines Blockgraphikadventures wird schon einmal vor dem Problem gestanden haben, daß durch viele Eingaben des Spielers die Grafik per Scrolling vom Bildschirm verschwindet. Professionelle Abenteuerspiele haben ein Textfenster. Das Scrolling ist auf den unteren Bereich des Bildschirmes beschränkt. Wie ich auf einer Wanderung durch das ROM-Listing festgestellt habe, kann man diesen Effekt durch das Ändern von nur einer Speicherstelle erreichen. Es ist dies die Adresse 59639. Da sie je-

doch im Kernelrom liegt, muß man zunächst die Bereiche von \$A000 bis \$BFFF und \$E000 bis \$FFF ins RAM kopieren. Danach kann man das ROM mit POKE 1,53 abschalten.
Genau dies geschieht in Listing 1. Da das Kopieren in BASIC etwa 70 Sekunden dauert, wird es in einer kleinen Maschinenroutine erledigt, die in Zeile 70 in den Kassettenpuffer geschrieben wird. Die Datasettenbesitzer können beruhigt sein, sie wird nur einmal benötigt. Die Routine wird mit SYS 828 gestartet (Zeile 80). Nach etwa einer halben Sekunde ist sie fertig. Danach kann das ROM mit POKE 1,53 ausgeblendet und die Scrollroutine verändert werden. Nehmen Sie also bitte die Nummer der ersten Zeile, ab der gescrollt werden darf (die Zeilen von 0 bis 24 gezählt), subtrahieren Sie davon 1 und poken Sie diesen Wert in die Adresse 59639 ein. Im Listing 1 wird ab der 21. Zeile gescrollt, also 21 minus 1 gleich 20 einpoken. Den Normalzustand stellen sie mit POKE 1,55 wieder her (das ROM wird wieder eingeschaltet).
Nun könnte der Wunsch auftreten, Text- und Grafikfenster getrennt zu löschen. Hierzu dienen die beiden Maschinenroutinen, die in Listing 2 eingepoket werden. Durch SYS 828 löschen Sie den scrollenden, durch SYS 842 den feststehenden Bereich des Bildschirms. Für Kassettenbenutzer und alle, die den Kassettenpuffer, den diese Routinen belegen, brauchen, sei hier erwähnt, daß diese Programme auch in anderen Bereichen lauffähig sind. Sie können sie also zum Beispiel auch in den Spriteblock 11 einlesen. Dann müssen Sie nur die Zeile 70 in Listing 2 wie folgt ändern: FOR I = 704 TO 704+26: ... Die Routinen werden dann mit SYS 704 und SYS 704+14 aufgerufen. Eine andere Möglichkeit wäre, die Routinen in den freien Bereich ab \$C000 zu legen. Dann müssen Sie statt 704 den Wert 49152 einsetzen. Gleiches gilt übrigens **nicht** für Listing 1. Diese Routine ist nur im Kassettenpuffer ab 828 lauffähig.
Kleine Erläuterung für Maschinensprachekundige:
In der Scrollroutine ab \$E8EA wird das X-Register mit 255 geladen. Danach wird es jeweils mit INX um eins erhöht und danach das Unterprogramm „Scrollen einer Zeile" aufgerufen. Dies geschieht solange, bis es den Wert 24 angenommen hat. Ändert man nun den Ladebefehl am Anfang um und weist dem X-Register einen höheren Wert zu, überspringt die Routine beim Scrollen die ersten Zeilen. Dies ist der ganze Trick.

Knut Dietrich

```
5 SZ=20:REM ==> SCROLLEN AB 4.LETZTER
                ZEILE
10 REM ************************
20 REM ** SCROLLMODIFIKATION **
30 REM **     (C)1986  BY      **
40 REM **   KNUT  DIETRICH     **
50 REM ************************
60 REM ==> EINLESEN DER MASCHINENROUTINE
           ZUM KOPIEREN VOM ROM INS RAM
70 FORI=828TO862:READX:POKEI,X:S=S+X:NEX
TI:IFS<>5231THENPRINT"FEHLER!!!":END
80 SYS828:REM ==> ROM INS RAM KOPIEREN
90 POKE1,53:REM ==> KERNEL- UND BASICROM
                 AUSSCHALTEN
100 POKE59639,SZ:REM ==> 1. SCROLLZEILE
                       MINUS 1 POKEN
110 DATA169,000,133,251,169,160,032,074,
003,169,224,076,074,003,133,252,160,000
120 DATA162,032,177,251,145,251,200,208,
249,202,240,004,230,252,208,242,096
```

Listing 1

```
10 SYS9*4096;ASSEMBLERAUFRUF
20 .OPT P,00;AUSGABE AUF BILDSCHIRM
30 *= 828;STARTADRESSE
40 ;
50 ;KOPIEREN VON KERNEL- UND BASIC-ROM
60 ;INS DARUNTERLIEGENDE RAM
70 ;
80 START LDA #$00;ZEIGER AUF $A000
90 STA $FB;(BASIC-ROM)
100 LDA #$A0
110 JSR SCHLEIFE;8 KBYTE INS RAM KOPIERE
N
120 LDA #$E0;ZEIGER AUF $E000  (KERNEL)
130 JMP SCHLEIFE;8 KBYTE KOPIEREN
140 SCHLEIFE STA $FC;HI-BYTE SPEICHERN
150 LDY #0;32*256=8192 BYTES=8 KBYTE
160 LDX #32;INS RAM KOPIEREN
170 LABEL1 LDA ($FB),Y
180 STA ($FB),Y
190 INY
200 BNE LABEL1;256 BYTES KOPIEREN
210 DEX
220 BEQ LABEL2;32 MAL
230 INC $FC;HI-BYTE UM EINS ERHOEHEN
240 BNE LABEL1;UNBEDINGTER SPRUNG
250 LABEL2 RTS;ZURUECK INS BASIC
```

Assembler-Listing zu Listing 1

```
10 REM *********************
20 REM ** CLR - ROUTINEN **
30 REM **  (C) 1986  BY  **
40 REM ** KNUT  DIETRICH **
50 REM *********************
60 REM ==> MASCHINENPRG EINLESEN
70 FORI=828TO854:READX:POKEI,X:S=S+X:NEX
TI:IFS<>4277THENPRINT"FEHLER!!!"
80 :
90 REM UNTEREN BEREICH LOESCHEN: SYS828
100 REM OBEREN BEREICH LOESCHEN: SYS842
110 :
120 DATA174,247,232,232,224,025,176,005,
032,255,233,048,246,096,174
130 DATA247,232,232,202,048,005,032,255,
233,048,248,096
653 BLOCKS FREE.
```

Listing 2

```
10 SYS9*4096;AUFRUF DES ASSEMBLERS
20 .OPT P,00;AUFGABE AUF BILDSCHIRM
30 *= 828;STARTADRESSE
40 ;
50 ;SCROLLBEREICH LOESCHEN
60 ;
70 CLEARSB LDX 59639;1. SCROLLZEILE LADE
N
80 CLSB1 INX;ZAEHLER ERHOEHEN
90 CPX #25;SCHON LETZTE ZEILE
100 BCS CLSB2;JA
110 JSR $E9FF;NEIN-ZEILE LOESCHEN
120 BMI CLSB1;UNBEDINGTER SPRUNG-WEITER
130 CLSB2 RTS;ZURUECK INS BASIC
140 ;
150 ;FESTSTEHENDEN BEREICH LOESCHEN
160 ;
180 CLEARFB LDX 59639;1.SCROLLZEILE LADE
N
190 INX;ZAEHLER RICHTIGSTELLEN
200 CLFB1 DEX;ZAEHLER ERNIEDRIGEN
210 BMI CLFB2;SCHON ALLE ZEILEN
220 JSR $E9FF;NEIN.ZEILE LOESCHEN
230 BMI CLFB1;UNBEDINGTER SPRUNG-WEITER
240 CLFB2 RTS;ZURUECK INS BASIC
```

Assembler-Listing zu Listing 2

Joystick-IRQ

Eine interruptgesteuerte Joystickabfrage für den C64, mit der ein Sprite gesteuert werden kann. Zwei Spriteblöcke wechseln sich ab, so daß eine Animation entsteht.

Die meisten BASIC-Programme werden durch die Joystickabfrage und Spriteanimation sehr langsam. Aus diesem Grund habe ich diese Interruptroutine in Maschinensprache geschrieben. Die Routine läuft parallel zu einem Programm, das dadurch nicht langsamer wird. Bei der Animation wechseln sich jeweils, für jede Richtung, Spriteblöcke ab. Es müssen also insgesamt acht Spriteblöcke vorher definiert sein. Folgende Blöcke sind vorgesehen:

Sprite nach oben: 190, 191
Sprite nach unten: 192, 193
Sprite nach links: 194, 195
Sprite nach rechts: 196, 197

Die Joystickabfrage ist für Port 2 vorgesehen. Vor dem Start der Routine mit SYS 49152 muß der erste Sprite eingeschaltet sein. Der Sprite läßt sich über den gesamten Bildschirm bewegen. Der Feuerknopf des Joysticks wird von dem Programm nicht beachtet. Nachdem man den Lader eingegeben hat, kann man das Programm als reinen MC-Code auf Diskette abspeichern und im eigenen Programm durch die Zeile

1 IF A=0 THEN A=1: LOAD"JOY-IRQ",8,8

am Anfang des Programms einladen. Eine zweite Möglichkeit wäre, die Daten in das Programm einzubauen. Vorteile bringt die Routine besonders beim Spielen. Sie kann aber auch gut in Menüs oder Zeichenprogrammen verwendet werden.

Sven Hasselmeyer

```
1 FORI=0TO185:READA:POKEI+49152,A:M=M+A:
NEXT
2 IFM<>23387THENPRINT"FEHLER IN DEN DATE
N."
4 PRINT"WOLLEN SIE DAS PROGRAMM ALS MC-C
ODE-    ROUTINE SPEICHERN ?"
6 GETA$:IFA$="N"THENEND
8 IFA$<>"J"THEN6
10 SYS(57812)"JOY-IRQ",8:POKE193,0:POKE1
94,192:POKE174,186:POKE175,192:SYS62957
32 DATA120,169,13,141,20,3,169,192,141,2
1
33 DATA3,88,96,173,0,220,170,41,1,208,27
34 DATA206,1,208,164,159,192,10,208,18,1
72
35 DATA248,7,192,190,208,6,238,248,7,76
36 DATA48,192,169,190,141,248,7,138,41,2
37 DATA208,27,238,1,208,164,159,192,10,2
08
38 DATA18,172,248,7,192,192,208,6,238,24
8
39 DATA7,76,80,192,169,192,141,248,7,138
40 DATA41,4,208,37,206,0,208,208,8,173,1
6
41 DATA208,73,1,141,16,208,164,159,192
42 DATA10,208,18,172,248,7,192,194,208,6
43 DATA238,248,7,76,122,192,169,194,141
44 DATA248,7,138,41,8,208,37,238,0,208,2
08
45 DATA8,173,16,208,73,1,141,16,208,164
46 DATA159,192,10,208,18,172,248,7,192,1
96
47 DATA208,6,238,248,7,76,164,192,169
48 DATA196,141,248,7,164,159,192,10,208,
4
49 DATA169,0,133,159,230,159,76,49,234
50 DATA1,141,16,208,76,49,234
```

BASIC-Listing zu Joystick-IRQ

```
5 REM DEMO
7 IFA=0THENA=1:LOAD"JOY-IRQ",8,8
10 FORI=0TO511:READA:POKEI+190*64,A:NEXT
:REM DATEN FUER SPR.LESEN
12 POKE53269,1:POKE53248,150:POKE53249,1
50
14 POKE2040,190:REM DATEN AUS BLK.190
16 SYS49152:REM IRQ-ROUTINE AKTIVIEREN
18 END
99 REM SPRITE-DATEN
100 DATA0,0,0,1,192,0,3,231,0,7,231,128,
15,231,192,15
102 DATA231,224,30,231,224,28,103,240,30
,231,240,31,231,240,31,231
104 DATA240,31,231,240,31,207,240,31,223
,240,15,255,224,15,255,224
106 DATA7,255,192,3,255,128,1,255,0,0,0,
0,0,0,0,0:REM DATEN FUER BLK.190
108 DATA0,0,0,1,192,0,3,247,0,7,247,128,
15,247,192,15
110 DATA247,224,30,247,224,28,119,240,30
,247,240,31,247,240,31,247
112 DATA240,31,247,240,31,255,240,31,255
,240,15,255,224,15,255,224
114 DATA7,255,192,3,255,128,1,255,0,0,0,
0,0,0,0,0:REM DATEN FUER BLK.191
116 DATA0,0,0,0,0,0,255,128,1,255,192,
3,255,224,7
118 DATA255,240,7,255,240,15,251,248,15,
243,248,15,231,248,15,231
120 DATA248,15,231,248,15,231,120,15,230
,56,7,231,120,7,231,240
122 DATA3,231,240,1,231,224,0,231,192,0,
3,128,0,0,0,0:REM DATEN FUER BLK.192
124 DATA0,0,0,0,0,0,255,128,1,255,192,
3,255,224,7
126 DATA255,240,7,255,240,15,255,248,15,
255,248,15,239,248,15,239
128 DATA248,15,239,248,15,239,120,15,238
,56,7,239,120,7,239,240
130 DATA3,239,240,1,239,224,0,239,192,0,
3,128,0,0,0,0:REM DATEN FUER BLK.193
132 DATA0,0,0,0,0,0,0,0,0,0,0,0,255,0,
3
134 DATA255,192,15,191,240,31,31,248,63,
191,252,63,255,252,31,252
136 DATA252,0,1,252,0,3,252,31,255,252,1
5,255,248,7,255,240
138 DATA1,255,192,0,127,0,0,0,0,0,0,0,
0,0,0:REM DATEN FUER BLK.194
140 DATA0,0,0,0,0,0,0,0,0,0,0,0,255,0,
3
142 DATA255,192,15,191,240,31,31,248,63,
191,252,63,255,252,63,255
144 DATA252,63,255,252,0,3,252,31,255,25
2,15,255,248,7,255,240
146 DATA1,255,192,0,127,0,0,0,0,0,0,0,
0,0,0:REM DATEN FUER BLK.195
148 DATA0,0,0,0,0,0,0,0,0,0,0,0,255,0,
3
150 DATA255,192,15,253,240,31,248,248,63
,253,252,63,255,252,63,63
152 DATA248,63,128,0,63,192,0,63,255,248
,31,255,248,15,255,240
154 DATA3,255,192,0,255,0,0,0,0,0,0,0,
0,0,0:REM DATEN FUER BLK.196
156 DATA0,0,0,0,0,0,0,0,0,0,0,0,255,0,
3
158 DATA255,192,15,253,240,31,248,248,63
,253,252,63,255,252,63,255
160 DATA252,63,255,252,63,192,0,63,255,2
48,31,255,240,15,255,224
162 DATA3,255,128,0,254,0,0,0,0,0,0,0,
0,0,0:REM DATEN FUER BLK.197
```

Demo-Listing zu Joystick-IRQ

DATA-Generator

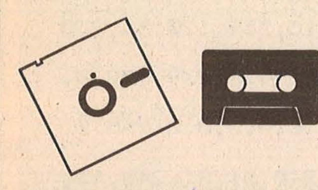
Hat man ein Assembler-Programm geschrieben und möchte es nun in ein BASIC-Programm einbinden, ist es sehr mühsam, den Maschinencode irgendwie in DATA-Zeilen zu bringen. Der „DATA-Generator" schafft hier Abhilfe und legt bestimmte Speicherbereiche automatisch in DATA-Zeilen ab (C128).

Nach dem Start mit „RUN 60000" sind folgende Informationen anzugeben:
1. ANFANGSADRESSE (0–65535)
2. ENDADRESSE (0–65535)
3. SPEICHERBANK (0–15)
4. ERSTE ZEILENNUMMER (0–60000 beziehungsweise >60200)
5. SCHRITTWEITE (beliebig)
6. ANZAHL (1–37 beziehungsweise 1–49)
7. (H)EX ODER (D)EZ ("H" oder "D")

(Die „ANZAHL" gibt die Menge der Zahlen in einer Programmzeile an.)
Sind alle Parameter korrekt angegeben, werden die DATA-Zeilen in DEZIMAL beziehungsweise HEXADEZIMAL generiert.
Danach kann die Routine mit „DELETE 60000–60200" gelöscht werden.

Uwe Komoß

```
60000 SCNCLR
60010 INPUT "ANFANGSADRESSE      ";A
60020 INPUT "ENDADRESSE          ";E
60030 INPUT "SPEICHERBANK        ";B
60040 PRINT
60050 INPUT "ERSTE ZEILENNUMMER  ";Z
60060 INPUT "SCHRITTWEITE        ";W
60070 INPUT "ANZAHL              ";S
60080 PRINT
60090 INPUT "(H)EX ODER (D)EZ   ";M$
60100 IF S>49 AND M$="H" OR S>37 AND M$=
"D" OR A>E OR Z=60000 OR M$<>"H" AND M$<
>"D" OR W=0 THEN PRINT CHR$(19);:GOTO 60
010
60110 BANK B
60120 FORI=1TO4:WI(I)=PEEK(227+I):NEXT
60130 CHAR,0,12,CHR$(27)+"T":TRAP60200
60140 SCNCLR
60150 IF M$="H" THEN BEGIN:PRINT Z"DATA
";:FOR I=1 TO S:PRINT RIGHT$(HEX$(PEEK(A
+I-1)),2)",";:IF A+I=E+1 THEN 60200:ELSE
:NEXT:PRINT CHR$(157)" "
60160 BEND:ELSE PRINT Z"DATA ";:FOR I=1
TO S:PRINT USING"###";PEEK(A+I-1);:PRINT
",";:IF A+I=E+1 THEN 60200:ELSE NEXT:PRI
NT CHR$(157)" "
60170 PRINT "A="A+S":Z="Z+W":GOTO 60140"
60180 POKE 208,0:POKE 842,19:POKE 843,13
:POKE 844,13:POKE 208,3
60190 END
60200 PRINT CHR$(157)" ":PRINT "FORI=1TO
4:POKE227+I,WI(I):NEXT:CHAR,0,12,CHR$(27
)+CHR$(64)":GOTO 60180
```



Kassetten- und Diskettenservice

Reversi/Directory-Druck/Verbesserte TRACE-Funktion/Formatieren unter BASIC
Schneider-K69
Kassette 21,80 Mark
Schneider-D69
Diskette 33,80 Mark
Printer-Spooler
Atari ST-D69
Diskette 22,80 Mark
Easy-DATA/REM-Killer/Page-Lister/Directory-Loader/Sicher laden
Atari XL-K69
Kassette 20,80 Mark
Atari XL-D69
Diskette 25,80 Mark
ZX-ärgert mich (16 + 48K)/UDG + (48K)
Spectrum-K69
Kassette 16,80 Mark
DATA-Generator
C128-K69
Kassette 12,80 Mark
C128-D69
Diskette 17,80 Mark
SCROLL & CLEAR/Joystick-IRQ
C64-K69
Kassette 16,80 Mark
C64-D-69
Diskette 21,80 Mark

Top-Games
Commodore 64 (Januar bis August 1985)
Schatzsuche/Roulette/Goblin 64/Reversi/Lifegame/River Raid 64/Schießbude/Chaser/Alien/Willi
Kassette C64-K596 39,– Mark
Diskette C64-D596 39,– Mark
Schneider CPC (Juli bis November 1985)
Vier gewinnt/Kamikaze/Köttel/Kniffel/The Wall/Pagoden von Peking/Car-Ware/Snake
Kassette CPC-K596 39,– Mark
Diskette CPC-D596 (3") 49,– Mark
Atari (Juli bis November 1985)
Golden Cellar/Sabotage/The Castle/Treasure Hunt/Lost in the Antartica/Mr. Pac/Höhlenflieger/Segelflug
Kassette Atari-K596 39,– Mark
Diskette Atari-D596 39,– Mark
MSX
Pacman/Super Memory/Monkey
Kassette MSX-K596 25,– Mark
Diskette MSX-D596 (3,5") 35,– Mark
Spectrum (April bis Dezember 1985)
3D-Golf/Frutti/Olympiade/Superbingo/Intellecto/Sechsundsechzig/Der Spion/Brücke/Labyrinth
Kassette Spectrum-K596 39,– Mark

Small Business und Utilitys
Commodore 64 (Januar bis Dezember 1985)
Master-Tape/Kalender/Sechs Stimmen/Disksort/Sprite de Luxe/Filemanager/Diskettendoktor
Kassette C64-K597 39,– Mark
Diskette C64-D597 39,– Mark
Game-BASIC/Macro-Assembler-Editor
Kassette C64-K595 39,– Mark
Diskette C64-D595 39,– Mark
Schneider (August bis Dezember 1985)
Druckerroutinen/Terminkalender/Astronomie/Dateiverwaltung/CPC-Hardcopy/Zeichen malen
Kassette CPC-K597 39,– Mark
Diskette CPC-D597 (3") 49,– Mark
Atari (August bis Dezember 1985)
Monitor/Sounddemo/Grafikdemo/Kalender
Kassette Atari-K597 29,– Mark
Diskette Atari-D597 29,– Mark
MSX (Oktober bis Dezember 1985)
Logo-Interpreter/Diagramm/Haushaltskasse
Kassette MSX-K597 25,– Mark
Diskette MSX-D597 (3,5") 35,– Mark
Spectrum (Mai bis November 1985)
Laufschrift/Super-DATA-Generator/Variablen-Lister/Weltenbummler/Spectrum Data/Super-Sprite
Kassette Spectrum-K597 39,– Mark

Bücher zum Commodore 64

VOGEL Computerbücher

Bradbury, A.J.
Das Abenteuer-Programmierbuch für den Commodore 64
Erst programmieren – dann spielen
196 Seiten, 18 Abb., 30,– DM
ISBN 3-8023-0809-3

Senftleben, Dietrich
Start mit Commodore-Logo
Das kleine Logo-Einmaleins
Grafik · Text · Musik
212 Seiten, 69 Abb., 30,– DM
ISBN 3-8023-0802-6

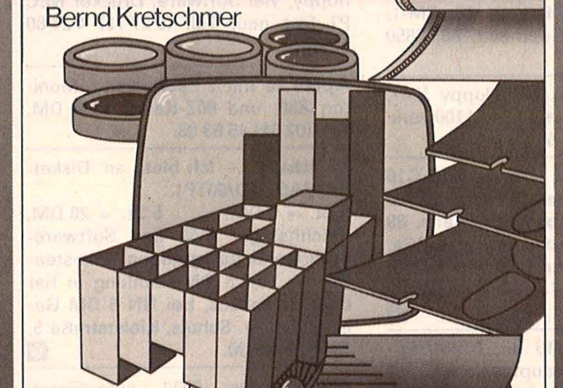
Wittwehr, Clemens
Spiel und Aktion mit Commodore-Logo
Mit der Schildkröte ins Land der Abenteuer
160 Seiten, 42 Abb., 28,– DM
ISBN 3-8023-0851-4

Sinclair, Ian
Mach mehr aus Deinem Commodore 64
Einführung in die Maschinensprache
180 Seiten, 69 Abb., 30,– DM
ISBN 3-8023-0808-5

Baumann, Rüdeger
Grafik mit dem Home-Computer
Grundlagen und Anwendungen programmiert in BASIC
328 S., zahlr. Abb., 38,– DM
ISBN 3-8023-0769-0

Rügheimer, Hannes
Spanik, Christian
Mein zweites Commodore-64-Buch
Das Buch das nach dem Handbuch kommt
280 Seiten, 23 Abb., 38,– DM
ISBN 3-8023-0808-5

Sacht, Hans-Joachim
Home-Computer kurz und bündig
Was jeder über Home-Computer wissen muß
152 Seiten, 72 Abb., 20,– DM
ISBN 3-8023-0790-9



Alfred Görgens
Was Drucker und Plotter alles können
Praktische Anwendungen mit Home- und Personalcomputern
CHIP WISSEN

VOGEL-BUCHVERLAG WÜRZBURG

VOGEL-Computerbücher helfen lernen, verstehen, anwenden

Sie erhalten bei Ihrem Buch- und Computerfachhändler kostenlos das neue Verzeichnis **„VOGEL-Computerbücher '85/86"** mit rund 100 aktuellen Titeln unserer Reihen **CHIP WISSEN** und **HC – Mein Home-Computer.**

Kretschmer, Bernd
Multiplan auf dem Commodore 64
Eine systematische Einführung
176 Seiten, 61 Abbildungen
Diese systematische Einführung in das Tabellenkalkulationsprogramm Multiplan in Deutsch hilft Ihren Commodore 64 für völlig neue Aufgaben einzusetzen. Dieses einführende Buch zeigt an einfachen Beispielen (Prozentrechnung, Umsatzstatistik, Textverarbeitung, Lieferschein), wie man mit Zahlen, Texte und Dateien verarbeiten und mischen kann.
ISBN 3-8023-0799-2 **28,– DM**

Görgens, Alfred
Was Drucker und Plotter alles können
Praktische Anwendungen
136 Seiten, 47 Abbildungen
Mit dieser Übersicht erhalten Sie die wichtigsten Informationen der gängigsten Druckertypen, ihre Besonderheiten und Leistungsmerkmale. Praktische Anwendungen mit Matrix-, Nadel-, Typenraddruckern, Laserdruckern sowie Trommel- und Flachbettplottern werden vorrangig behandelt.
ISBN 3-8023-0783-6 **28,– DM**

HC Mein Home-Computer

...Kaufen ... Verkaufen ... Tauschen ... Kontakte knüpfen ...

Bitte verwenden Sie den vorbereiteten Auftragscoupon für Ihre Anzeige in der HC-Börse. Sie finden ihn auf den nächsten Seiten.

G = gewerblich

Biete an Hardware

Atari 130 XE, Disk 1050, Touch-Tablet, 12 Games, 5 Bücher, usw. VB 750 DM. Tel. (0 69) 43 33 03.

Apple IIe, 80 Zeichen + 64K, 1 Lw., Literatur u. Disketten, VB 1700 DM. Tel. (02 02) 64 51 58, ab 16.30 h.

C64-Farbmonitor, 1701-1641-MPS801, zu verkaufen. Tel. (0 22 24) 8 07 37.

Plantron PC, 256-KB-RAM, 2 Disk., Lw. à 360 KB, nur 1850 DM! Schwenkb. Monitor, 498 DM; B.System MS-DOS 2.11, 198 DM; Software f. PC: Fibu — Lohn — Faktura, nur 798 DM. Versand p. Nachnahme + Porto. Firma Ernst Pieber, Am Bellershof 1, 4150 Krefeld 1. **G**

CPC464, CP664, CPC6128 User! Lichtgriffel mit Programmen und dt. Anleitung nur 49 DM. Versand gegen Scheck/Nachnahme, Info gratis! Fa. Schißlbauer, Postfach 11 71S, 8458 Sulzbach, Tel. (0 96 61) 65 92 bis 21 Uhr. **G**

Sharp MZ 821, Neuw. Software, Joystick 370 DM, Tel. (0 98 76) 3 51.

Dragon 32, Software (u.a. Systemerw.) + Kass.rec., Drucker GP 100, Lit. 300 DM. Tel. (0 70 22) 3 47 55.

ZX81, 16K, Aufs.tast. u. z. Einb., Lichtgr., Steckk. dazu (Pl. f. PIO), Drucker, Literatur, Software, Zubeh., VB 500 DM, Tel. (06 81) 6 73 00.

Apple IIc, neues Demogerät, Garantie, 1250 DM. Tel. (0 98 76) 3 51.

●●●●●●●●●●●●●●●●●

ZX-Spectrum
Reparatur-Schnelldienst
und Ersatzteile
Computer & Medientechnik
Heinz Meyer
Rahserstr. 52, 4060 Viersen 1
Tel. (0 21 62) 2 29 64. **G**

●●●●●●●●●●●●●●●●●

Sinclair QL dt. Vers., 128KB mit 720 KB Disk inkl. Software, Lit., 1200 DM, Tel. (0 48 41) 6 45 91.

Österreich: CPC-464, Farbmonitor, Floppy + 6 3"-Discs, CP/M 2.2, 2 Joysticks, CPC-464, Intern, Preis 8000 öS, Tel. (00 43 77 29) 4 68.

IBM Graphics Printer/Epson FX 80, VHB 700 DM. Tel. (04 31) 64 21 40.

Casio PB-700, OR-4, FA-10, CM-1, Sw. Mahrholz, Goethestr. 86, 2850 Bremerhaven.

C 64 m. Hypra, Load, Floppy 1541, Monitor 1701, Lektüre, VB 1100 verk. Tel. (09 31) 70 94 56

C16 C16 Besitzer Achtung **C16 C16** 64KByte Speichererweiterung nur 149 DM, bei Einbau durch uns. 89 DM bei Selbsteinbau. Info gratis! Fa. Klaus Schißlbauer, Postfach 11 71E, 8458 Sulzbach, Tel. (0 96 61) 65 92 bis 21 Uhr. **G**

NCR DM V, 8/16 Bit, (CPM/MS-DOS) 256KB-Hauptspeicher, 2 Disk.-Laufwerke, 1 freistehende erweiterbare Festplatte 10 MB, 1 Druckerinterface, 1 Textverarbeitung (TEX-ASS), Tel. (0 56 92) 23 09.

★★★★★★★★★★★★★★★★
★ **Drucker MPP 1361** f. CBM ★
★ 8032-8296 150 Z./sek. à 550 ★
★ DM. Tel. (0 61 57) 8 31 68. ★
★★★★★★★★★★★★★★★★

Apple 2+, 64K, Z80, 80Z, 2 LW Monitor, IBM-Tast. CP/M, dBase, Wordstar, Cobol usw. 1500 DM. Tel. (0 24 06) 7 98 83. Mo—Fr 15—18.30 Uhr.

Apple II Europlus, 64K, 2 LW, 80 Z., RS 232C, EPSON-Karte, Monitor (bernstein), PASCAL u.a. SW, H6., VB 1800 DM; **Apple Graphic-Tablet** (orig.) mit Contr., SW, H6. VB 700 DM; F. Meyer, Tel. (0 89) 68 38 58.

ATARI 260 ST, Komplettsystem + Softw. + Literatur, 1850 DM. J. Brunk, Tel. (0 82 33) 9 22 39.

EPSON PX-8, kaum gebraucht, 1450 DM. Tel. (02 34) 53 13 36.

Supermac Mac Plus mit 2 MB, 8500 DM. Externe Festplatte Magnet 20, 5200 DM (alles so gut wie neu). Tel. (0 40) 54 53 42.

Tandy, Modell III, Neuzustand, 2 LW, mit RS 232 (evtl. eig. Software). VB 1200 DM. Tel. (0 69) 50 14 09 (ab 19.00 Uhr).

Atari 1040 STF, Uhr, Roms, Zweitfloppy, viel Software, Drucker NEC P7 fast neu, Tel. (0 27 72) 4 24 80 abends.

Apple 2e mit 2 Laufwerken, Monitor, Z80- und 80Z-Karte. 1600 DM. Tel. (02 31) 45 53 08.

!Achtung! — Ich biete an: Disketten BASF 2D/96TP1.
1 St. = 5 DM; 5 St. = 20 DM.
Reichhaltiger HW- und Software-Programmkatalog gegen Unkostenbeitrag von 5 DM. Zahlung in bar oder Barscheck, bei NN 5 DM Gebühr. Bei M. Schulz, Kleiststraße 5, 1000 Berlin 30. **G**

Commodore: 8032, aufgerüstet, Floppy 8250 LP + TR-Drucker HR 15. VB: 5500 DM. Tel. (0 21 04) 78 12.

Apple II + original! 80 Z Mont. Handbu. VB 1550 DM. Tel. (0 69) 15 33 81 91, Zi 150!

Wegen Neuanschaffung verkaufen wir unser **MC**-Entwicklungssystem **PMDSII** von Philips incl. Software und diversem Zubehör (Z80 und 8085 Emulatoren) zu einem außerordentlich günstigen Preis. Institut für Prozeßtechnik. Tel. (07 21) 6 08-45 25 od. 45 20. **G**

Sharp MZ 731-1A, io. Tel. (07 11) 4 20 22 31.

Kaypro 4/84, incl. Software 2500 DM VHB. Tel. (0 61 33) 5 05 25.

IBM-PC-XT 2 Disk., Laufwerke + 10 MB, Platte, 512 KB a. Speicher, Monochr., Farbbildschirm, Drucker, etc., etc., VB 8200 DM. Chiffre 092572.

Alphatronic P3, 2 Laufw. à 760 KB, 2500 DM. Tel. (0 20 54) 8 32 75.

Commodore C64/C128/VC20 User! Lichtgriffel mit Programmen und dt. Anleitung nur 49 DM. Versand gegen Scheck/Nachnahme. Fordern Sie unseren kostenlosen Commodorezubehörprospekt an. Firma Klaus Schißlbauer, Postfach 11 71C, 8458 Sulzbach, Tel. (0 96 61) 65 92 bis 21 Uhr. **G**

XT-komp. mit 2 AT-Disk + Harddisk Farbmonitor etc., VB 16 000, Apple 2 komp. 2 Disk, Z80, 80Z, FDC-Kontroller, Stahlbl. Gehäuse, etc ., SW VB 2200, Harddiskkontroller für Apple 2, (Frank + Britting) VB 900 DM. Tel. (0 76 21) 4 65 78.

Apricot PC, 2 × 720K 9"-Mon. DOS 211 WS, DB, SW, SC nur privat gebr., wg. Syswechsel, DM 5900. Tel. (04 21) 32 31 80.

MAG PC 256, zu verk., IBM-kompat. 256K, Color Mono, Karte, 1 FD 360K, Dt. Tastatur nur 2500 DM. Tel. (0 51 54) 14 95, Lutz verlangen!

Schneider 6128, Drucker, Handbüch., Disket. 1A-Zust. Tel. (0 93 23) 37 49.

CBM SFD 1001, 1 MB formatiert mit C64/128-Interface, kompl. Garantie 11/86, (0 96 21) 8 15 40 oder (09 41) 3 19 64. 695,— DM.

TA PC8+ orig. Lit. da Zeitmangel, zu verk. VB 400 DM. Tel. (0 69) 88 40 06.

IBM Portable PPC, 7 Monate alt m. 576KB-RAM, 2 × 360 KB-Laufwerke V24 und Parallel-**Schnittstelle,** zus. VB 6450 DM, dazu diverse Programme. Tel. (0 81 22) 78 19.

CBM 3032 + 3022 mit Notstrom für 30 Min., Schallschluckgeh., Diskcontr. für 2 ★ 400K eingeb., VHB 700 DM. Tel. (0 91 31) 30 17 62. **G**

Reko Electronic
Computer-Systeme, MSX-Hard-/Software und Zubehör. Reinhard Korfmann, Postfach 32 24, 5810 Witten 3. **G**

HC-BÖRSE

Biete an Hardware

Apple III incl. Monitor 2. Laufw. Parall-K.CP/M-K. Apple II-Emula., MBASIC Visicalc usw., 3000 DM. Tel. (0 41 01) 6 16 67.

HP 110, Portable, Lotus 1-2-3, Diskettenlw. Matrixdrucker, neuwertig, VB 4500 DM. Tel. (0 89) 16 16 77.

Sony Hit Bit + Sanyo Farbmon. CD 3195A zu verkaufen. Tel. (0 72 51) 45 45.

Apple II mit Language-Card, 2 LW, Druckerinterface, Monitor, Pascal, Handb., 2000 DM. Tel. (0 89) 71 88 03.

PX8, 1450 DM, M1D. Tel. (05 11) 8 09 21 61.

Apple 2+ Komp. 3 LW (143/640 KB), 80Z, Z80, 2 Contr. Uhr, IBM-Geh. Mon. SW, Lit., Preis VB. Tel. (02 28) 61 17 12.

Atari 600XL/800XL/130XE User! Lichtgriffel mit Programmen und dt. Anleitung nur 49 DM. Versand gegen Scheck/Nachnahme. Info gratis! Fa. Schißlbauer, Postfach 11 71A, 8458 Sulzbach, Tel. (0 96 61) 65 92 bis 21 Uhr. G

IBM-Kompatible, komplett, 1999 DM. Tel. (0 23 51) 7 82 21, ab 18 h + Sa/So. G

Sirius 1 m. 2 × 12 MB-Floppy u. div. Software (Wordstar, dBase I, Videothekprogr.), Preis VB. Tel. (05 71) 5 73 53.

■■■■■■■■■■■■■■■■
■ **Computerkauf leicht gemacht** ■
■ Wir finanzieren Ihren Computer und Zubehör. Info anfordern! Auch für Händler interessant. ■
■ **SKG Bank, Postfach 321, Cecilienstr. 4, 6600 Saarbrücken, Tel. (06 81) 3 03 01 11.** G ■
■■■■■■■■■■■■■■■■

Disketten
$5^1/_4$" 48 tpi **1,99 DM**
$3^1/_2$" 135 tpi **4,65 DM**
auch andere, 6 Mon. Garant. Allg. Austro-AG, Ringstr. 10, D-8057 Echingen, Tel. (0 81 33) 61 16. G

C16/116, RAM-Erweiterung, 64 KByte, interner Umbau, Ports bleiben frei ★ Basic V 3.5 60671 Free ★ Computer mit 100 DM einsenden an: Uwe Peters, Tannenweg 9, D-2351 Trappenkamp. 1 Woche Ret. Preis, incl. Mat. Arb. Vers.kost.

■ **64-KByte-Speicher f. C16** ■
Bausatz/Fertig für **84/131 DM, Einbau-Service** f. C116 & 16, **87 DM,**
IEC-Bus f. C116, 16 + 4; C64, C128; **RS232** f. C116 & C16.
Infos von **SAS-Bernd,** Langgasse 93, **5216** Niederkassel 5, **Tel.** (02 28) 45 26 26.
■ **und mehr für Commodore** ■ G

★★★★★★★★★★★★★★★★
★ **CBM-Drucker,** MPP 1361, 150 Z./s. Breitwagen z. Höchstgebot. Tel. (0 61 57) 8 31 68. ★
★★★★★★★★★★★★★★★★
★★★★★★★★★★★★★★★★
★ **Compaq Desk** pro 20-MB-Platte, 360-KB-Diskette, 25-MB-Band-Laufwerk, parallel, serielle Schnittst., Monitor bernstein. Dreh-Tastatur, 9900 DM. Tel. (0 61 57) 8 31 68. ★
★★★★★★★★★★★★★★★★

Apple IIe, 128K, Duodisk, Monitor, Akustikkoppler, Interf., 802, 280, Clock, IEEE 488, Centr.-Interface, Joystick, Softw. + Lit., kompl. 2850 DM. Tel. (0 30) 6 62 27 12.

Amiga, 256K, 1 + 2 MB Speichererw., Floppy, Software. Tel. (0 60 31) 6 19 50.

Altos 8000-10E, ADDS, 3200 DM, Altos 8000-6S, ACS 8000-6SE-, F. Platte, Streamer, Visual 200, 3600 DM; Altos 5-5, 10 MB, Visual 200, 2900 DM, Tel. (0 61 32) 5 89 24. G

■■■■■■■■■■■■■■■■
■ **Schweiz: MCS-Disketten,** made in Switzerland, die Besten, alle Formate zu Top-Preisen bei Krummenacher-Computer, CH-Tel. (01) 7 40 29 77. ■
■ G ■
■■■■■■■■■■■■■■■■

PC-Zubehör, preiswert. Tel. (0 73 61) 6 43 64. G

Biete an Software

520 ST Prg-Verkauf Info 80 Pf, U. Ton, Kantstr. 35, 4005 Meerbusch.

Haustechnische Programme für Sharp PC 1260 und kompatible Heizlast-Rohrnetz-Heizkörper, Druckausdehnungsgefäße, Info bei Ego, Postfach 138, 1210 Wien.

Sharp PC-1500 (A): Extra-RAM i. ROM, Bereich neue Befehle, Soft- u. Hardware-Erw., kostenl. Info, Ch. Germelmann, Pf. 345, 3422 Bad Lauterberg 1

Cobol-Compiler, Alphatronic P3 600 DM, Tel. (0 20 54) 8 32 75.

Cobol-Compiler für Apple III 250 DM, Tel. (0 20 54) 8 32 75.

Headline das universelle Druck- und Gestaltungsprogramm für Matrixdrucker! Tel. (07 21) 88 76 21. G

Maskengenerator für alle Turbo-Programmierer. MS-DOS-DEMO 20,– ab 19 Uhr Tel. (0 52 42) 3 57 60.

dBase-Anwendungen. Tel. (0 23 51) 7 82 21. G

––– **Grafikprogramme** –––
Für C 64 und IBM.
Verschiedene Programme zur Auswahl. Ausführliche Beschreibung und Demodiskette gegen 10,– DM in bar bei M. Schulz, Kleiststr. 5, 1000 Berlin 30. G

Biete an Software

Schneider CPC. Gratisinfo bei Friedr. Neuper, PF 72, 8473 Pfreimd. G

● Können **Sie** Schreibmaschine schreiben? **Keymaster,** der Schreibmaschinenkurs für den CPC 464. Deutscher Zeichensatz, 56 KB Programmlänge, 49 DM + Porto. U. Keim, An d. Waldesruh 5, 6400 Fulda-Niesig.

NEU NEU NEU NEU NEU
Hacker-Buch nur 38 DM bei VS bei NN + Gebühr. Info gegen 80 Pf. Oppermann, Walldürner W. 24, 1000 Berlin 20.
NEU NEU NEU NEU NEU

CBM 80XX: Spitzenprogramme aller Art! Gratisliste bei Thomas Liedtke, 7140 Ludwigsburg, E.-Bälz-Str. 17.

MZ-700/800 Besitzer! Fordern Sie unseren Softwareexclusivkatalog an. Natürlich gratis! Nur bei AM Technologies c/o A. Mielke, Vinnhorster Weg 35, 3000 Hannover 21. Superpreise! (Händler). G

BZ-Homöopathie für Commod. 128 – auch für den an Naturheilkunde inter. Laien sehr wertvoll. (auch Vers. f. C64 lieferb.), kennen Sie schon unsere prof. Software (C64) aus den Ber. Astrologie, Biorhythmik, Heilpraktikerprüf. u. Persönlichkeitsanalyse? Info v. Beate Zille, Berger Str. 272, 6000 Frankfurt 60. G

★ ★ **Schneider-Software** ★ ★ preisw. Spiele, Mathe, Anwend. (z.B. Minivisicalc), Katalog g. 1 DM von Schneidersoft, Wagner, Gartenstr. 4, 8201 Neubeuern.

Wärmebedarfberechn. 4701/83
K-Zahlberechnung DIN 4701/83
Rohrnetzberechnung-Programm
Druckausdehnung
Für VC64 + 1541 + Drucker, je 100 DM, vom Fachmann privat. Tel. (0 40) 6 72 46 46.

Amiga Software
Public Domain Software für Amiga aus den USA, je Disk 20 DM. Kopierservice Public Domain Software Christian Bellingrath; Trift 10, 5860 Iserlohn. Tel. (0 23 71) 2 41 92. G

Sharp-Pocket, **Computer**-Bücher. Bestellen Sie sich passend zum Rechner Ihre Literatur. Fordern Sie noch heute für 80 Pf Rückporto eine Liste der gesamten Angebote. Oppermann, Walldürner W. 24, 1000 Berlin 20.

Public Domain Software aus USA
679 Disks für IBM
246 Disks für CPM
88 Disks für C 64
10 Disks für C 128
10 Disks für Amiga
Preise: IBM je 15 DM, CPM u. C64 je 10 DM, C 128 je 12 DM, Amiga je 20 DM, günstige Mengenrabattstaffeln, z.B. 88 C64 Disks 600 DM, 200 PC-Blue 1500 DM. Katalog auf Diskette für IBM (3 Disks) 25 DM, CPM Katalog auf Disk 10 DM, C64/128/Amiga Liste 80 Pf. Rückporto. Kopierservice Public Domain Software. Christian Bellingrath, Trift 10, 5860 Iserlohn, Tel. (0 23 71) 2 41 92, Telex 8 27 937. G

C64, C128, C16/116, VC20. Neue ernsthafte Programme, z.B. Textverarbeitung, Katalog gegen 2 × 80 Pf in Briefm., Computerservice T. Hofstede, A. d. Windmühle 8, 5010 Bergheim. G

Für Commodore 8000:
Programmierer-Softwaretools u.a. Hilfsprogramme abzugeben. Chiffre 092562.

CBM 8096-SK/8250 mit SW. Lagerverwaltung, Buchhaltung, Textverarbeitung und Kalkulat. einzeln oder zus. Chiffre 092560.

PC + CP/M + Frei-SW. Tel. (0 23 51) 7 82 21,18 h. G

Suche Hardware

CBM-IBM-Hardware sucht 0407607333.

Suche Epson-FX80-Dru + Fischertechnik für CBM-IBM. 0407607333.

Sharp MZ 80 A + 80 FB gesucht! Tel. (0 44 31) 12 12.

Suche **Kartenleser**/Lochkartenleser (auch defekt). J. Steltner, Allmersstr. 10, 3000 Hannover 1, Tel. (05 11) 88 12 10.

SX-64 gesucht. Angebote an Eberhard Müller, Wisentweg 5, 7000 Stuttgart 31, Tel. (07 11) 86 15 64.

ATARI-Software für XL/XE-Rechner. Viels. Anwendungen, enorm preisw. C/D. Kostenl. Info bei H. Kegel, Am Erlenberg 13, 6107 Reinheim.

Suche Software

Suche Hausverwaltungsprogramm f. Commodore PC 10, IBM komp. Tel. (0 60 32) 8 56 10).

Auftragscoupon für Kleinanzeigen in HC-BÖRSE

gezielt und kostengünstig

■ kaufen
■ verkaufen
■ tauschen
■ Kontakte knüpfen

Gewerbliche Gelegenheitsanzeigen je Druckzeile 11,50 DM zuzügl. MwSt.

Private Gelegenheitsanzeigen je Druckzeile 7,50 DM inkl. MwSt.

Bitte ausschneiden (fotokopieren) und ausgefüllt an HC-Börse, Postfach 67 40, 8700 Würzburg schicken!

Lesernummer

Absender

Vor- und Zuname

Beruf

Straße und Nr.

Wohnort

PLZ

Bitte veröffentlichen Sie nebenstehenden Text von ______ Zeilen à ______ DM in der nächsterreichbaren Ausgabe von **HC**

Bei Angeboten:
Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

Unterschrift Datum

Bitte veröffentlichen Sie in der nächsterreichbaren Ausgabe nachstehenden Text:

30 Buchstaben je Zeile, inkl. Satzzeichen und Zwischenräumen, bei normaler Schrift. Bei Fettdruck, grafischen Zeichen usw. müssen wir uns Abweichungen vorbehalten.

Gewerbliche Anzeigen werden mit G gekennzeichnet.

☐ Biete an ☐ Suche ☐ Verschiedenes
☐ Hardware ☐ Hardware ☐ Kontakte
☐ Software ☐ Software

Chiffregebühr 6 DM inkl. MwSt.

Suche Software

Software vom TI-59 im IBM-PC? Suche Soft- u. Hardware um Modul- u. Magnetkartenprogramme in GW-Basic zu transferieren. B. Heuchert, Teichredder 9, 2351 Husberg, Tel. (0 43 21) 2 87 99.

S. Fibu f. CBM 3032. Tel. (0 40) 7 96 42 13.

Fibu für Sirius 1, bis 500 DM. Tel. (05 71) 5 73 53.

Suche Apple-Works mit Beschreibung. Tel. (02 03) 43 42 03.

★ Programme gesucht: ★
Für eine Veröffentlichung suchen wir gegen Honorar Programme, vorzugsweise Grafik-Anwendungen sowie Nutzprogramme mit Pfiff für Schule, Beruf und Freizeit — für
★ Sharp MZ 700/800
★ Epson HX 20
★ UCSD- und Turbo-Pascal
Die besten Chancen, gut honoriert und publiziert zu werden, haben auf Datenträger eingereichte Programme mit ausführlicher Beschreibung. Einsenden an: Vogel-Verlag, z. Hd. Herrn Armin Schwarz, Schillerstr. 23a, 8000 München 2. G

Chinesisch und/oder **Japanisch** f. Studienzwecke von privat gesucht: Sprach- bzw. Schriftlehrprogramme einschließlich Textprogramm (KANZI/KANJI/KANA). Erläuterung auf Engl. angenehm. Besitze SX 64/ Epson GX80, würde Hardware zukaufen. Musterausdr! An Postfach 15, CH-4059 Basel.

Suche Kontakt zu Amiga-Usern, auch Anfängern, zwecks Hard-Software u. Infotausch. Tel. (0 60 31) 9 13 01.

„Hilfe"
Gezocht: Het Adres van de Gebruikers Groep van de Sharp MZ-700. Sturen Naar: P. v. Klaveren, Grovestins 10, 7608 HM Almelo, Nederland.

Kontakte

PC Software erstellt: M. Morgenroth, Tel. (02 02) 71 34 12.

Achtung Amiga-Fans! Suche Kontakte im Großraum 4300 Essen. Bis dann! Rainer. Chiffre: 092575.

Amiga-Einsteiger sucht Kontakt zwecks gegenseitiger Hilfe. R. Doliwa, 4 Düsseldorf 1, Heresbachstr. 2, Tel. (02 11) 34 92 39.

Olivetti M20 Freaks für Multiplan, Oliword u. Faktura gesucht. Wer kann PCOS/BASIC programmieren? M20 Software anbieten. Raum Ffm. Unterstützung gegen Std.-Lohn f. meine M20 FDN/HDN gesucht. Wer kann Modem-Übertragungssoftware zw. 2 M24 anbieten oder entwicekeln? Tel. (0 69) 39 80 44 u. 31 82 10.

Sharp Hisoft Pascal Klub e.V.
Wir Pascalieren!
Egal ob mit Hisoft- oder Turbo-Pascal, ob Anfänger oder Kenner, wir bieten Interessierten ein Forum zum Erfahrungsaustausch an. Unsere Klubzeitschrift berichtet stets aktuell und behandelt auch u.a. Basic u. Assembler mit, dazu gibt es neue Sharp-Informationen. Gegen 3 DM in Briefm. erhalten Sie ein Probeexemplar zugesandt. Bei Fragen stets Porto beifügen. SHKe. V., V. Petersen, Behringstr. 23, 2000 Hamburg 50, Tel.: (0 40) 3 90 14 05.

Verschiedenes

Anrufbeantworter o. FTZ-Nr.
398 DM
Anrufbeantworter m. FTZ-Nr.
698 DM
per NN Reising, Füllengarten 52, 6600 Saarbrücken 5, Tel. (06 81) 7 94 89.

Commodore-Service-Manual's für alle Typen liefert ab sofort: Schaltungsdienst Lange Berlin, PF 47 06 53, D-1000 Berlin 47, Tel. (0 30) 6 03 20 03, Telex 1 84 339. G

HC 11/83—11/85 gg. Gebot! A. Zander, Mainzer Str. 4, 1000 Berlin 31.

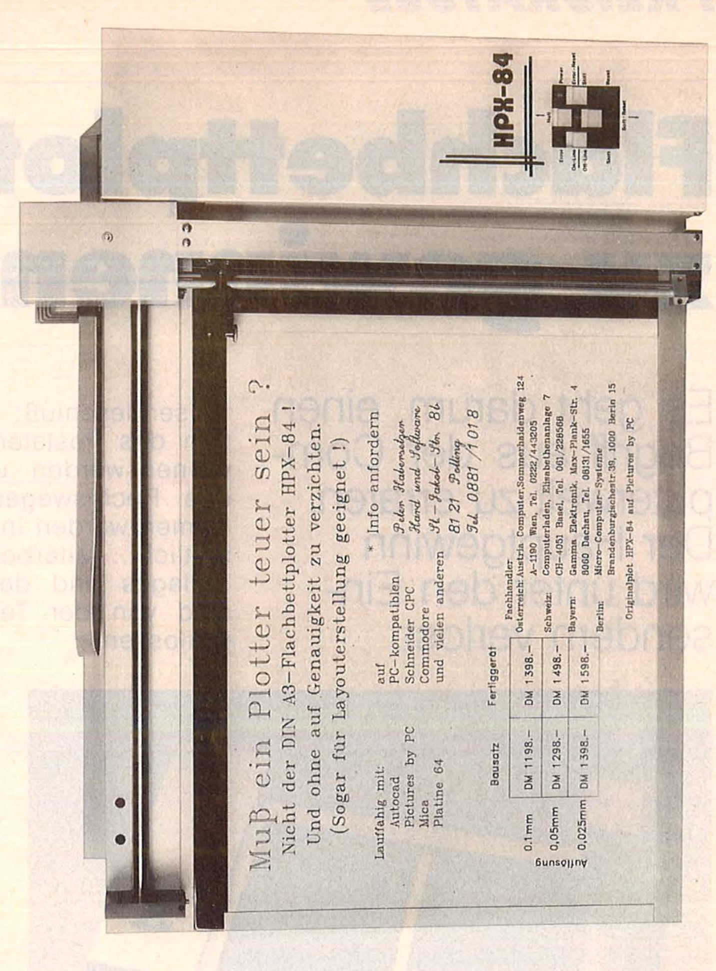
Preiswerte Hard-/Software für Home- und Personal-Computer. K & N, Pf. 90 08 06, 2100 Hamburg, Tel. (0 40) 7 63 13 65. G

Programmierung von Tools- und Anwendungen in allen Sprachen. Vehrenberg, Blücherstr. 42, 4000 Düsseldorf 30, Tel. (02 11) 44 60 18.

∗ CP/M-Tips gesucht ∗
Helfen Sie uns, die Geheimnisse des Betriebssystems CP/M zu lüften. Gesucht sind gegen Honorar Tips, Tricks, Utilities und Background-Informationen in Form von Kurzbeiträgen mit Listing auf Datenträger, lauffähig auf einem gängigen Rechner.
Einsenden an: Vogel-Verlag, z. Hd. Herrn Armin Schwarz, Stichwort: CP/M-Special, Schillerstr. 23a, 8000 München 2. G

∗ MS-DOS-Tips gesucht ∗
Helfen Sie uns, die Geheimnisse des Betriebssystems MS-DOS zu lüften. Gesucht sind gegen Honorar Tips, Tricks, Utilities und Background-Informationen in Form von Kurzbeiträgen mit Listing auf Datenträger, lauffähig auf einem gängigen Rechner.
Einsenden an: Vogel-Verlag, z. Hd. Herrn Armin Schwarz, Stichwort: MS-DOS-Special, Schillerstr. 23a, 8000 München 2. G

Bausatzkatalog ∗ 300 Seiten ∗ gratis ∗ Porto + Vers., 6 DM. Liebherr elect., 8353 Osterhofen. G



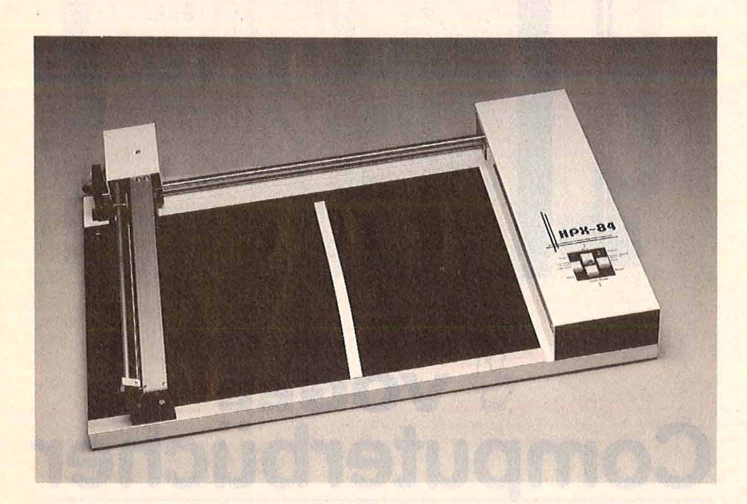
Flachbettplotter HPX-84 zu gewinnen

Es geht darum, einen Begriff aus der Computerwelt zu erraten. Der Hauptgewinn wird unter den Einsendern verlost.

Einsendeschluß: 15. 9. 1986 (Datum des Poststempels). Die Gewinner werden unter Ausschluß des Rechtsweges ermittelt. Die Namen werden in HC 12/86 veröffentlicht. Mitarbeiter des Vogel-Verlages und deren Angehörige sind von der Teilnahme ausgeschlossen.

Die Preise

Zu gewinnen gibt es einen HPX-84-Flachbettplotter sowie zehn interessante Bücher aus der Welt der Mikrocomputer und Elektronik.



Der Flachbettplotter HPX-84 für den Gewinner dieses Rätsels: präzise und leise.

1								
2								
3								
4								
5								
6								
7								
8								
9								

Wir haben uns eine knifflige Aufgabe für Sie ausgedacht. In die waagerechten Reihen sind Wörter einzutragen. Nach richtiger Lösung steht in der mittleren Senkrechten ein Begriff aus der Datenverarbeitung. Schreiben Sie das Lösungswort auf eine Postkarte, und senden Sie diese an:
HC-Redaktion
Kennwort: Plotter
8000 München 100

Die Fragen

1. Himmelsgewölbe
2. Rindslendenstück
3. Wissenschaftlicher Mitarbeiter
4. Feinste Tonware
5. Ind. Kaste
6. Waldfrucht
7. Verstand
8. Europ. Hauptstadt
9. Regierungsform

Das bietet der HPX-84-Plotter

– Centronics-Schnittstelle
– Bis 32 767 horizontale und vertikale Schnitte
– Fertige Befehle für Rechtecke und Kreise
– Papiergröße bis DIN A3

Der Plotter wurde von der Firma Peter Habersetzer gestiftet.

Die Auflösung des Ariola-Preisrätsels:

Eine Glücksfee hat uns aus den vielen richtigen Einsendungen zum Preisrätsel aus HC 6/86 den Hauptgewinner und die Gewinner der zehn Buchpreise gezogen. Die richtige Lösung heißt:

KONRAD ZUSE.

Der 1. Preis, ein Spielepaket von Ariolasoft, geht an:

Norman Reil
Augsburger Str. 58/2
8034 Germering 2

Die zehn Buchpreise erhalten:
Günter Bormes, 5523 Röllersdorf
Detlef Bußmann, 4600 Dortmund 30
Elli Hiller, 7032 Sindelfingen
Gerhard Höllisch, 8952 Marktoberdorf
Michael Jung, 4750 Unna-Königsborn
Olaf Kunkel, 4100 Duisburg 25
Ferdi Ried, 5330 Königswinter 1
Hugo Schulz, 2000 Hamburg 62
Annegret Weinberger-Groß, 7147 Hochdorf
Wolfgang Zupp, 3012 Langenhagen

Herzlichen Glückwunsch!

DATA BECKER Buchhits zu Commodore C16 und Plus/4



Mit diesem Buch zu Ihrem C16 verfügen Sie über eine leichtverständliche Einführung in Handhabung, Einsatz und Programmierung des C16, die keinerlei Vorkenntnisse voraussetzt. Themen: Bedienung von Tastatur und Editor, erster Befehl und erstes Programm, BASIC-Einführung mit Erstellung einer kompletten Adressenverwaltung! Nur der richtige Einstieg garantiert den späteren Erfolg!
C16 für Einsteiger, 205 Seiten, DM 29,–

Haben Sie einen C16/116 und kein Futter für ihn? Dann kann Ihnen mit diesem Buch geholfen werden. Aus dem Inhalt: Spiele, Malprogramm, Laufschrift, Textverarbeitung, Dateiverwaltung, Vokabeltrainer, Text-Hardcopy, Merge, Shape-Editor, simulierter Direktmodus, der integrierte Monitor, Zeropage, Routinen des Betriebssystems und des BASIC-Interpreters.
Dieses Buch gehört griffbereit neben Ihren Rechner.
C16 Tips & Tricks, 201 Seiten, DM 29,–



Klar und ausführlich werden die Möglichkeiten der Grafikprogrammierung auf den Rechnern C16/C116/Plus/4 vorgestellt. Die Grafikbefehle des BASIC 3.5 werden mit vielen Beispielen verdeutlicht: Farb-, Multi-, Color- und Hi-Res-Modi, Befehle zur Steuerung von Grafik und Shapes, 2D-/3D-Programmierung, CAD und Statistik. Mit diesem Buch lernen Sie Ihren Rechner von seiner stärksten Seite kennen.
Das Grafikbuch zu C16 · C116 · Plus/4, ca. 300 Seiten, DM 29,–. Erscheint ca. August

Machen Sie mehr aus Ihrem Rechner! Anhand vieler Programmbeispiele führt der Autor in den Befehlssatz der Rechner C16/C116/Plus/4 ein: Ein-/Ausgabe mit Input/Print, Rechnen mit Variablen, Stringverarbeitung, Verzweigung und Schleifen, Grafik- und Musikprogrammierung, komplette Befehls-Übersicht. Für Einsteiger und Fortgeschrittene gleichermaßen interessant.
Das BASIC-Buch zu C16 · C116 · Plus/4, ca. 250 Seiten, DM 29,–. Erscheint ca. August

Programmierung in Maschinensprache – leichtgemacht. Der Befehlssatz des Prozessors, die Verwendung des TED-MON, die wichtigsten Routinen des Betriebssystems. Auf diese Weise können Sie das Know-how, welches in Ihrem Betriebssystem steckt, in Ihre eigenen Programme integrieren.
C16 · C116 · Plus/4 Maschinensprache, ca. 300 Seiten, DM 29,–. Erscheint ca. August



Wer seinen Commodore Plus/4 richtig nutzen will, der kommt an diesem Ideenbuch nicht vorbei. Aus dem Inhalt: Was ist eine Textverarbeitung, welche Vorteile bietet sie, Installation der Plus 4-Software, Serienbriefe, Steuerzeichen, Peeks und Pokes, Balkengrafik, Funktionsplotter, Dateiverwaltung, Listing der Zero-Page (Systemspeicher), u.v.m. Gehört zu jedem Plus/4!
Effektiv & Kreativ mit dem Commodore Plus/4, 244 Seiten, DM 49,–

Commodore Plus/4 Tips & Tricks enthält eine hochkarätige Sammlung von Anregungen, Ideen und fertigen Lösungen zur Programmierung Ihres Plus/4: Anwenderprogramme aus den Bereichen Unterhaltung, Grafik, Text- und Dateiverwaltung. Viele Utilities wie Text-Hardcopy, REM-Killer, Mergen, Shape-Editor und Datumsberechnung. Die wichtigsten Zeropageadressen und Betriebssystemroutinen führen hin zur Programmierung in Assembler. Eine echte Hilfe für alle ernsthaften Programmierer.
Plus/4 Tips & Tricks, 221 Seiten, DM 29,–

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

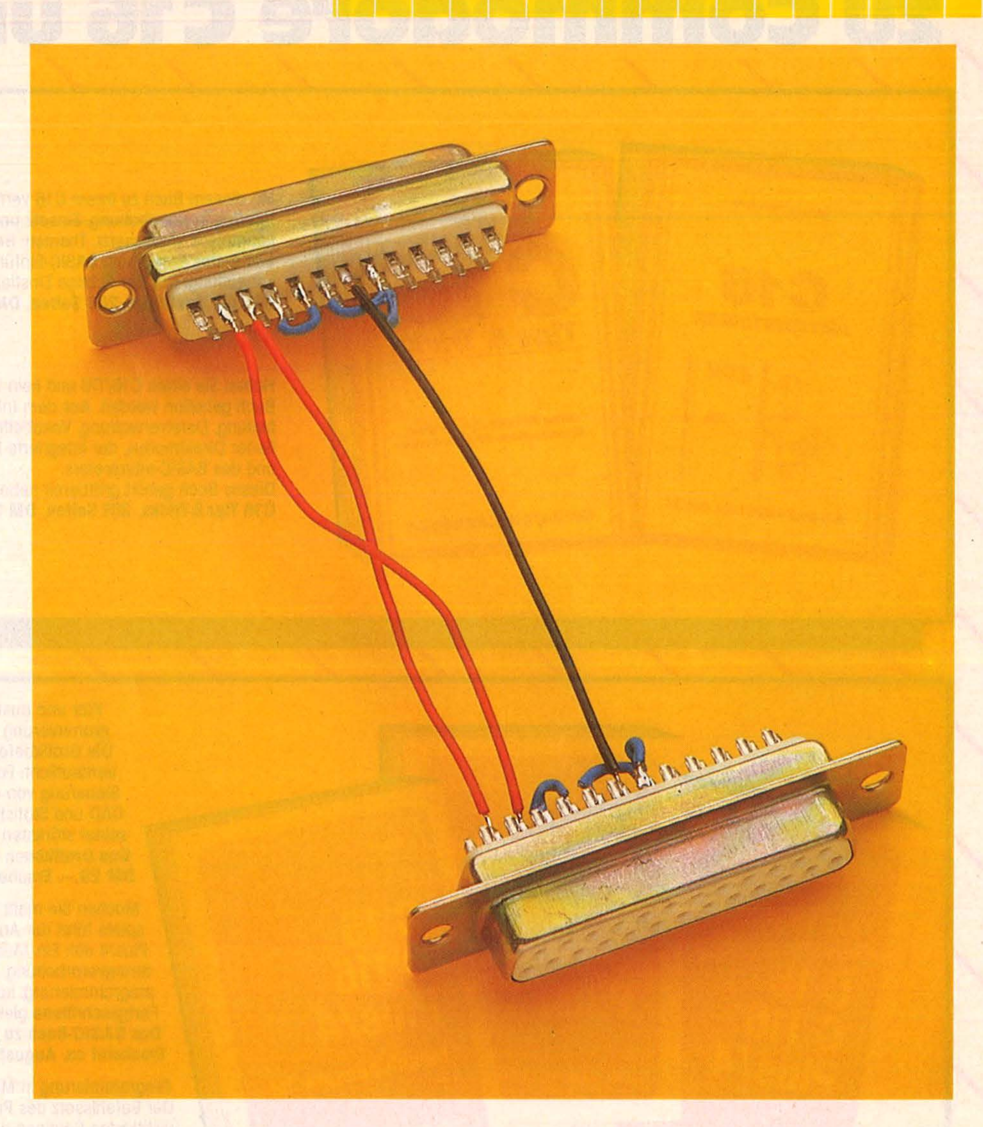
zzgl. DM 5,– Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name

Straße

Ort

Nullmodem: *Zwei 25polige Stekker und ein paar Litzenstücke sind alles, was man dafür braucht. Auf das „Gewußt-wie" kommt es an*



Ein Nullmodem macht es möglich: Datenaustausch zwischen verschiedenen Rechnern. Und ein Schnittstellentester kann Telefonkosten senken

NULLMODEM UND TESTER

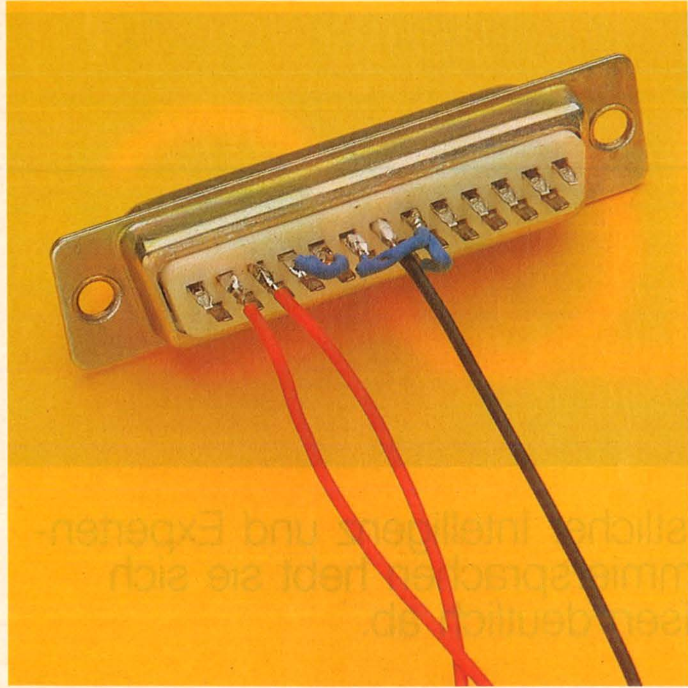
Ein Nullmodem ermöglicht es, zwei verschiedene Rechner über ihre serielle Schnittstelle zusammenzuhängen. Wenn auf beiden zum Beispiel ein Terminalprogramm läuft, können also Dateien vom C64 auf einen IBM-Rechner übertragen werden.
Als Material brauchen wir:

* zwei RS232C-Stecker (weiblich)
* Isolierte Litze
* Lötkolben, Lötzinn
* Messer zum Abisolieren
* Schraubenzieher

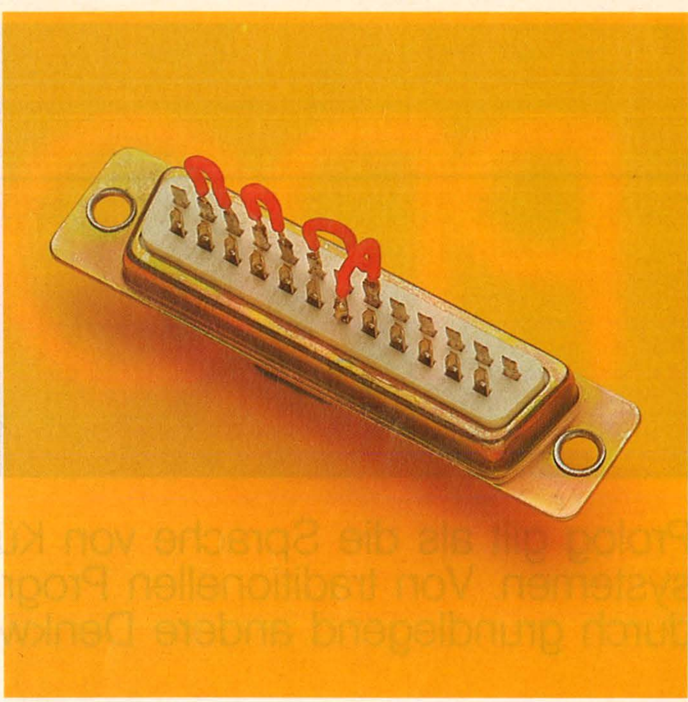
Drei Litzen, etwa zehn Zentimeter lang, werden auf beiden Seiten abisoliert und mit je einem Tropfen

Das Nullmodem überträgt Daten

Lötzinn verzinnt, damit sich keines der dünnen Kupferdrähtchen selbständig macht. Die beiden Stecker werden so gelegt, daß man die ins Plastik eingedruckten Numerierungen der Leitungen lesen kann. Auf beiden Steckern wird der Pin 7 (Masse) mit einem Tropfen Lötzinn versehen. Beide Masse-Pins werden mit einer Leitung miteinander verbunden. Nun kommen die beiden wichtigsten Pins der RS232C-Schnittstelle: Transmit DATA (TD) und Receive Data (RD), also Daten senden und empfangen. Es sind die Pins zwei und drei, die nun mit den beiden übriggebliebenen Leitungen kreuzweise verbunden



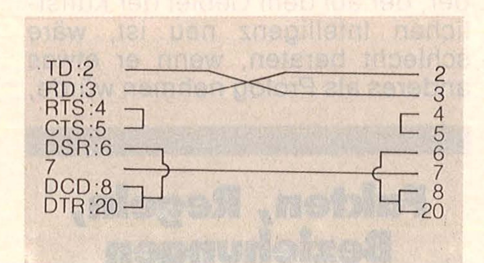
Verdrahtung aus der Nähe: *Beim Steckerkauf darauf achten, daß auf einen Anschluß mit Stiften natürlich ein (weiblicher) mit Buchsen gehört*



Selbstgespräche: *Nach dieser einfachen Lötarbeit können DFÜ- und Hackprogramme getestet werden, ohne daß Telefongebühren anfallen*

werden. Das heißt: Pin 2 am einen Stecker wird mit Pin 3 am anderen Stecker verbunden und umgekehrt. Die Leitung, auf der der erste Computer sendet, ist auf die Empfangsleitung des zweiten geschaltet (und umgekehrt).

Eigentlich ist unser Nullmodem schon fertig. Was jetzt nur noch fehlt, ist der „Handshake", also das Verfahren, mit dem die Computer sich gegenseitig mitteilen, wann wer empfangs- bzw. sendebereit ist. Dies dient dazu, Fehler auszuschließen und Datenverlust zu vermeiden. Da dieses Verfahren aber im Normalfall nicht verwendet wird, schließen wir die Leitungen kurz, indem wir jeden Computer den Handshake mit sich selber machen lassen. Dazu setzen wir zwischen die Pins vier und fünf mit kurzen Litzenstücken und ein paar Tropfen Lötzinn in dem Stecker eine Lötbrücke. Ebenfalls in jedem Stecker werden mit zwei kurzen, auf beiden Seiten abisolierten und verzinntem Draht die Pins sechs, acht und zwanzig miteinander verbunden. Vor dem Zusammenschrauben der Stecker muß jetzt nur noch die Funktionsfähigkeit unseres Nullmodems überprüft werden. Am besten sendet man einen längeren Text mit 2400 Baud hin und her. Gibt es Schwierigkeiten, dann sind entweder die Lötstellen nicht einwandfrei, oder eines der feinen Kupferhaare hat sich doch selbständig gemacht.



Schaltplan: *So sieht die Verdrahtung des Nullmodems aus*

Wer an Terminalprogrammen strickt oder Mailboxen entwickelt, wer Auto-Hacker programmiert und NUA-Scanner fabriziert, kennt das Problem: Bis das Programm das macht, was es soll, sind viele – sinnlose – Telefoneinheiten verbraten. Geld, das man besser anwenden kann. Zum Beispiel in einem Schnittstellenteststecker. An Material brauchen wir dafür:

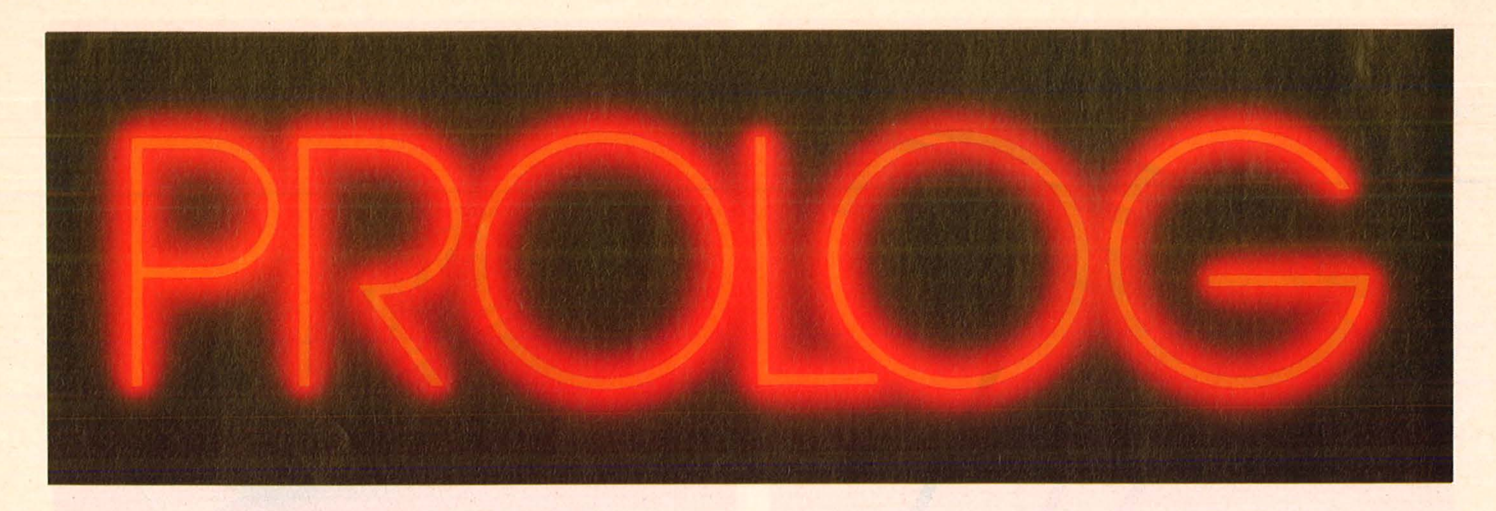
* einen RS232C-Stecker
* etwas isolierte Litze
* Lötkolben, Lötzinn
* Messer und Schraubenzieher
* eine Steckerhülle

Der Schnittstellentest erspart das Telefon

Ähnlich wie beim Nullmodem geht es bei unserem Schnittstellenteststecker darum, den Computer mit sich selber unterhalten zu lassen. Nur kommt jetzt dazu, daß die Daten, die der Computer aussendet, auch bei diesem wieder ankommen. Deshalb werden jetzt miteinander verbunden: Die Pins zwei und drei; die Pins vier und fünf und die Pins sechs, acht und zwanzig. Die nebeneinanderliegenden Pins werden wie bekannt mit einem gutplazierten Tropfen Lötzinn und einem Litzenstück verbunden, zwischen sechs, acht und zwanzig tun dieselbe Arbeit zwei kleine Drahtstücke. Ist unser Stekker fertig, sollten die Drähte nur noch mit einer Steckerhülle gegen unbeabsichtigtes Abreißen geschützt werden.

Beim C64, der ja bekanntlich keine serielle Schnittstelle hat, kann unser Stecker seine Arbeit aber an dem Modemkabel aus der vorletzten Ausgabe tun. Steckt man unseren Stecker hinein, so passiert folgendes: Jedes Zeichen, das getippt und damit über die Schnittstelle gesendet wird, wird über den Stecker zurückgeschickt und erscheint auf dem Bildschirm. Ein Autohacker oder NUA-Scanner kann so stundenlang und kostenlos ausprobiert werden, bis er auf jede DATEX-Meldung fehlerfrei funktioniert.

Joachim Graf



Prolog gilt als die Sprache von Künstlicher Intelligenz und Expertensystemen. Von traditionellen Programmiersprachen hebt sie sich durch grundlegend andere Denkweisen deutlich ab.

Mit der fünften Computergeneration ist vor allem die Hoffnung verbunden, daß Expertensysteme in den verschiedensten Bereichen Menschen auch auf intellektuellem Gebiet entlasten können. Ein Expertensystem kann als Hilfsmittel betrachtet werden, menschliche Fähigkeiten für ein bestimmtes Spezialistengebiet zu speichern und abzurufen. In Frage kommen zunächst unter anderem die Fehlersuche in technischen Geräten und Reparaturanleitungen, Diagnose von Krankheiten und Behandlungsvorschläge, Überwachungs- und Steueraufgaben, Vorhersage, Wissensvermittlung. Benötigt werden dabei Fähigkeiten wie logisches Schließen und der Umgang mit vagem Wissen und dessen Bewertung.

Typisch für Expertensysteme sind nicht mathematisch sichere Schlußfolgerungen wie ,,Wenn a >7, dann auch a >5", sondern Aussagen der Form ,,Wenn x wahr ist und y wahr ist, dann besteht die Wahrscheinlichkeit P, daß auch z wahr ist". Durch Aufdecken neuer Details und Heranziehen zusätzlichen Wissens kann eine Aussage dann entweder untermauert oder abgeschwächt werden. Ein ideales Expertensystem sollte dem Benutzer auch erklären können, wie eine bestimmte Folgerung zustande gekommen ist. Falsch ist es, Expertensysteme mit den Computern der fünften Generation zu identifizieren. Prototypen sind schon seit Jahren auf den Gebieten Medizin, Technik und molekulare Genetik im Einsatz.

Alex Goodall, geschäftsführender Direktor der Gesellschaft Expert Systems, zum Stellenwert der Programmiersprache Prolog: ,,Jeder, der auf dem Gebiet der Künstlichen Intelligenz neu ist, wäre schlecht beraten, wenn er etwas anderes als Prolog nehmen würde,

Fakten, Regeln, Beziehungen

da es einfacher anzuwenden ist als LISP und mächtiger als die andere Alternative Pop-2." Zwei weitere Sprachen werden in diesem Zusammenhang noch genannt: VALID und occam. Für sie alle gilt, daß sie eine Abkehr von den herkömmlichen Programmiermethoden bedeuten. In Prolog wird dies besonders deutlich.

Im Gegensatz zu traditionellen Programmiersprachen fehlen in Prolog fast ganz Sprachelemente, die den Programmablauf steuern sollen (GOTO, FOR-NEXT, usw.). Der Programmierer codiert nicht einen Algorithmus, sondern er gibt die Fakten, Beziehungen und Regeln an, die in der zu beschreibenden Problemwelt Gültigkeit haben. Im Hinblick auf Expertensysteme könnte man auch sagen: Wissen mit all seinen gegenseitigen Abhängigkeiten und Vernetzungen wird zusammengetragen.

Ein einfaches Beispiel, in ,,Turbo-Prolog" geschrieben:

```
domains
    personen, sport = symbol
predicates
    spielt(personen,sport)
clauses
    spielt(hans,fussball).
    spielt(susanne,volleyball).
    spielt(erich,schach).
    spielt(robert,tennis).
    spielt(uta,X) if spielt(erich,X).
```

Das Programm informiert darüber, welche Sportarten jeder einzelne einer kleinen Gruppe von Personen ausübt. Im ersten Teil des Programms (domains) werden alle Objektnamen aufgeführt, die beim betreffenden Problem eine Rolle spielen, zugleich wird mitgeteilt, von welchem Typ sie sind. In unserem Beispiel sind sowohl personen als auch sport vom Typ symbol. Weitere mögliche Typen sind zum Beispiel real und integer. Im zweiten Teil (predicates) werden die Namen von Beziehungen aufgeführt und die Objektnamen genannt, die dabei verknüpft sind. ,,spielt" stellt in unserem Beispiel eine Beziehung zwischen personen und sport dar. Damit ist der Deklarationsblock abgeschlossen. Es folgt der Prozedurteil.

Im Prozedurteil (clauses) werden zunächst vier Fakten mitgeteilt. Die fünfte Prozedur ist eine sogenannte Klausel, bestehend aus Kopf (Konklusion) und Rumpf (Prämisse). Sie besagt, daß jeder

Sport, der von Erich ausgeübt wird, auch von Uta ausgeübt wird. Statt if findet meist :– Verwendung, das die gleiche Bedeutung hat, also auch ,,falls" gelesen werden kann. Der Rumpf einer Prozedur ist meist wesentlich komplexer und besteht aus einzelnen Literalen, die durch logische Operatoren verknüpft sind. Die Zeichen , und ; stehen dabei für das logische UND bzw. das logische ODER. In der fünften Prozedur taucht die Variable X auf. Variable werden im Gegensatz zu Konstanten immer groß geschrieben. Während Konstante für einzelne Individuen oder Konzepte stehen, benutzt man Variable, wenn zu unterschiedlichen Zeitpunkten unterschiedliche Individuen oder Konzepte gekennzeichnet werden sollen.

An das compilierte und gestartete Programm können Anfragen gestellt werden. Die Aufforderung dazu wird in Turbo-Prolog in Form von ,,Goal:_" ausgegeben. Die Eingabe ,,spielt(robert,schach)." zum Beispiel quittiert das Programm mit der Bemerkung ,,True". ,,spielt(hans,tennis)." wird mit "no solution" beantwortet, denn aus dem Programm geht weder hervor, daß Hans Tennis spielt, noch, daß Hans kein Tennis spielt. Die Frage, ob Uta Schach spielt, ist schon nicht mehr ganz so offensichtlich zu entscheiden. Das Faktum 3 zusammen mit der Klausel am Ende des Programms führen jedoch zum Schluß ,,TRUE". Anfragen können auch Variable enthalten.

Cut stoppt Backtracking

,,spielt(I,schach)" zum Beispiel wird mit
,,I = erich
I = uta
2 Solutions"
beantwortet.

Eine Fragestellung wie ,,Welche Sportarten sind in der Gruppe alle vertreten?" ist ebenso möglich. Dem Fragesteller ist es hierbei anscheinend egal, wer jeweils welche Sportart ausführt. Die Anfrage an das Prologprogramm enthält eine sogenannte anonyme Variable, das Zeichen _. Die Anfrage wird in der Form spielt(_,S). eingegeben und führt zur Ausgabe
,,S = fussball
S = volleyball
S = schach
S = tennis
4 Solutions"

Bei so einfachen Programmen sind natürlich Ergebnisse leicht vorhersagbar, bei komplexeren Programmen nicht mehr. Die Auswertungsstrategie von Prolog bleibt jedoch immer die gleiche.

Jede Anfrage läßt sich als Hypothese auffassen. Das gesamte Wissen (Fakten, Beziehungen, Regeln) wird nun von Prolog nach einem Trial- und Error-Verfahren durchstöbert, um die Hypothese zu beweisen. Dabei wird ständig versucht, jeweils zwei Prologstrukturen durch geeignete Bindung von freien Variablen zu identifizieren. Während dies bei ,,spielt(I,fussball)" gelingt, weil die Variable I an die Konstante hans gebunden werden kann, mißlingt es bei ,,spielt(uta,volleyball)".

Die Suche erfolgt nach dem Backtrackingverfahren, das sehr zeitraubend sein kann. Jedoch lassen sich in die Prozeduren an geeigneten Stellen Cuts mit dem Zeichen ! einbauen, wenn dort eine weitere Suche sowieso aussichtslos ist. Bezeichnend für die Denkweise in Prolog ist es, daß es auf die Reihenfolge der Prozeduren im Programm überhaupt nicht ankommt. Sie kann beliebig umgestellt werden, ohne daß sich an der Arbeitsweise des Programmes irgendetwas ändert.

Obwohl natürlich auch Zuweisungen in Prolog nicht fehlen und ebenso Eingabe-, Ausgabe- und andere Routinen vorhanden sind, muß Prolog Programmierern in traditionellen Sprachen zunächst als Spezialsprache vorkommen. Die ungewohnte Denkweise läßt es unmöglich erscheinen, zum Beispiel auch Spiele effektiv programmieren zu können. Daß es dennoch möglich ist, noch dazu sehr kompakt und einigermaßen gut lesbar, zeigt das Programm-Listing ,,Türme von Hanoi" auf Seite 95.

An diesem Listing ist unter anderem auch zu erkennen, daß Rekursion in Prolog eine große Rolle spielt. Welche Gestalt Rekursion in Prolog annimmt, soll am Beispiel der Warteschleife am Anfang des Prozedurteils vom Programm ,,Türme von Hanoi" erläutert werden. Die Prozedur delay überprüft die Prämisse dd(100). 100 stellt eine Bindung der Variablen N an dd (N) dar. Das führt zur Zuweisung N1 = N – 1 und wegen dd(N1) zum rekursiven Aufruf der Prozedur dd. Die Rekursion wird erst gestoppt, wenn N1 = 0, denn dd(0):-!. hat einen Abbruch zur Folge. Die Länge der Pause kann natürlich variiert werden, wenn man 100 durch geeignete andere Zahlen ersetzt.

Stichworte

Anonyme Variable Die Variable ,,_", die anstelle einer gewöhnlichen Variable benutzt wird, wenn die Werte, die die gewöhnliche Variable annehmen könnte, nicht von Interesse sind.

Arbeits-File File, in dem das Prolog-Quell-Programm steht, um dann compiliert oder ausgeführt zu werden.

Argument Wie Parameter ein Sammelname für Objekte und Variable einer Relation.

Backtracking Eine Methode, bei der versucht wird, das vorherige Teilziel zu erfüllen, wenn das momentane nicht erfüllt werden kann.

clause Ein Faktum oder eine Regel für eine bestimmte Aussage.

Dialog-Fenster Das System-Fenster von Turbo-Prolog, in dem nach externen Zielen gefragt wird und das die Antwort des Prolog-Systems darauf präsentiert.

domain Programm-Teil, der über den Typ der Objekte informiert.

Editor-Fenster Das System-Fenster von Turbo-Prolog, in dem zum Beispiel Prolog-Programme eingegeben und bearbeitet werden.

Expertensystem Ein Computersystem, das die Fähigkeit eines Experten in einem bestimmten (meist kleinen) Bereich nachahmt.

externes Ziel Ziel, das vom Prolog-Programm angefordert wird und der Benutzer im Dialog eingeben muß.

Faktum Eine Beziehung zwischen Objekten. In spielt(hans,fussball) zum Beispiel ist ,,spielt" der Name der Beziehung und hans und fußball sind Objekte.

fail Ein Teilziel, das nicht erreicht werden kann.

freie Variable Variable, die momentan auf keinen bestimmten Wert festgelegt ist.

goal Die Menge aller Teilziele, die erfüllt werden sollen.

integer Eine Ganzzahl im Bereich von −32768 bis +32767.
internes Ziel Ziel, das im Programm im Zielteil formuliert ist.
Liste Eine besondere Objektart, Ansammlung von Objekten, die in eckige Klammern eingeschlossen und durch Kommas getrennt sind.
Operator-Priorität Hierarchie, mit der die Reihenfolge festgelegt wird, in der Operationen ausgeführt werden.
Parameter Sammelname für alle Objekte und Variable einer Relation.
predicate Jedes Faktum und jede Regel gehört zu einer Aussage (predicate), die den Namen der betreffenden Beziehung angibt und die Typen der verknüpften Objekte.
real Eine Dezimalzahl im Bereich von 10^{-307} bis 10^{+308}.
Rekursion Technik, bei der eine Prozedur sich zur eigenen Definition selbst aufruft.
Regel Beziehung zwischen einem Faktum und mehreren Teilzielen, die erfüllt sein müssen.
Stack Speicherbereich zum Parameteraustausch.
Standard-Predicate Predicate, bereits innerhalb des Turbo-Prolog-Systems definiert.
Symbol Ein Name beginnend mit einem Kleinbuchstaben.
Trace-Fenster Bei Turbo-Prolog kann man in diesem Fenster zu Testzwecken die Programmausführung genau verfolgen.
Variable Ein Name beginnend mit einem Großbuchstaben. Die Variable kann stellvertretend für den Wert eines bestimmten Objekts stehen.
Verbundziel Ziel, das aus mindestens zwei Teilzielen besteht.
Ziel-Baum Ein Diagramm der verschiedenen Möglichkeiten bei der Bewertung der Teilziele eines Gesamtzieles.

Prolog-Systeme, Literatur und ein Beispiel-Listing

Prolog-Systeme sind zur Zeit nur für einige Home- und Mikrocomputer erhältlich. Sie weichen in der Ausführung sehr stark voneinander ab. Am ausgereiftesten ist ,,Turbo-Prolog" von der Firma Heimsoeth in München. Es läuft auf dem IBM PC und allen dazu kompatiblen Rechnern. Ein didaktisch gut aufgebautes, ausführliches Handbuch mit vielen kurzen Prolog-Listings erleichtert sehr den Zugang zu dieser ungewöhnlichen Sprache.

Für den Commodore 64 ist ,,Prolog 64" von Brainware in Wiesbaden. Im Sinclair-Computershop in München ist Micro-Prolog für den Sinclair-Spectrum 48K erhältlich. Beide weisen untereinander und zum Turbo-Prolog starke Unterschiede auf. Es sind beides Interpreterversionen, während Turbo-Prolog eine Compilerversion ist. Der begrenzte Speicherplatz von C64 und Spectrum schränkt die Möglichkeiten beider Systeme von vornherein stark ein.

Ein Prolog-System mit dem Namen ,,Prolog II" kommt von Wolfgang Kreitmair und Manfred Krifka in Dachau. Es ist bereits für den Apple Macintosh erhältlich, arbeitet mit Fenstertechnik und hat einen Editor, der sich ähnlich wie der Mac-Write-Editor verhält. Es handelt sich um keine Compilerversion. Prolog II soll bald auch für IBM-PC/-XT/-AT und kompatible Rechner erhältlich sein. Innerhalb dieses Artikels wurde vorwiegend auf ,,Turbo-Prolog" aufgebaut.

Wer sich für das Umfeld der Sprache Prolog interessiert, dem



steht mit Büchern über Künstliche Intelligenz und Expertensysteme eine bereits jetzt sehr ansehnliche und ständig wachsende Auswahl an Computerliteratur zur Verfügung. Nur drei davon seien hier genannt. ,,Künstliche Intelligenz in BASIC" von Mike James bei mvg baut auf den Voraussetzungen auf, die BASIC-Programmierer mitbringen, und dringt über die Spieltheorie in die Materie ein, ohne dann wesentliche Aspekte auszulassen. Klare wissenschaftliche Sprache, manchmal allzu mathematisch abstrakt, zeichnen das Buch ,,Künstliche Intelligenz und Expertensysteme" von Stuart E. Savory im Oldenbourg Verlag aus. Es beinhaltet ein eigenes Kapitel über Prolog. Zahlreiche Literaturnachweise verschaffen einen nahezu lückenlosen Überblick über alle relevante Literatur. Einen Blick in die Zukunft wirft ,,Die fünfte Computergeneration" von G. L. Simons im Hanser-Verlag. Bücher, die sich ausschließlich und ausführlich mit Prolog befassen, ohne Handbücher einer bestimmten Prolog-Version zu sein, sind sehr rar. Eines der wenigen ist ,,Programming in Prolog" (in englischer Sprache) von W. F. Clocksin im Springer Verlag. Allerdings wirkt es im Aufbau auf Lernende nicht sehr motivierend.

Ein vollständiges Listing des Spieles ,,Türme von Hanoi" soll einen Einblick geben, wie ein etwas größeres Prolog-Programm aufgebaut ist. Es fällt auf, daß fast jede Prozedur rekursiv definiert ist und alle durch gegenseitige Aufrufe stark verzahnt sind. Die eigentliche Bewegung der Scheiben übernehmen die Prozeduren move_vert und move_horizon. In deren Rumpf befindet sich auch die Routine field_attr, sie ist eine Bibliotheksroutine und ist deshalb im Programm nicht definiert. Vertikale und horizontale Bewegung sind jeweils in zwei Prozeduren formuliert, entsprechend einer Bewegung nach oben oder unten bzw. rechts oder links. Wie bei der Warteschleife am Anfang des Prozedurteils setzen Cuts mit ! der rekursiv definierten Bewegung ein Ende, wenn sich an der Höhe (bzw. seitlichen Verschiebung) gar nichts ändert. Weitere Aus- und Eingabe-Routinen befinden sich am Ende der Prozedur hanoi; dort wird der Cursor in Ausgangsstellung gebracht, eine Aufforderung auf den Bildschirm geschrieben und die Tastatur gelesen, nachdem unter anderem mit der Bibliotheksroutine makewindow der Bildschirmaufbau zu Beginn des Spieles erfolgte. *-br*

Lernen am praktischen Beispiel: ein typisches Prolog-Listing

```
  /**************************************************************/
  /*                    TOWERS OF HANOI                         */
  /*                    ===============                         */
  /*              this hanoi is slowed down                     */
  /* Call the predicate hanoi with the number of slices, you want.*/
  /* Example:  Hanoi(6).                                        */
  /**************************************************************/
DOMAINS
  TIME, ROW, COL, NUMBER = INTEGER

PREDICATES
  hanoi( NUMBER )
  move(  NUMBER, NUMBER, ROW, ROW, ROW, COL, COL, COL )
  inform(  NUMBER, NUMBER, ROW, ROW, COL, COL )
  makepole( NUMBER, NUMBER, COL)
  delay()  dd(TIME)
  move_vert(COL,NUMBER,ROW,ROW)
  move_horizon(ROW,NUMBER,COL,COL)

CLAUSES
  delay :- dd(100).
  dd(0):-!.
  dd(N):-N1=N-1,dd(N1).

  hanoi(N) :-
        N<=13,!,
        VB=2+6*N,VH=3+N,CV=N, CM=3*N, CH=5*N,
        STCOL=(79-6*N)/2, STROW=(25-VH)/2,
        makewindow(1,7,7,"Hanoi",STROW,STCOL,VH,VB),
        makepole(N,N,CV),
        move(N,N,0,0,0,CV,CM,CH),
        cursor(0,0), write("Press any key"),readchar(_).

  hanoi(_):- write("maximum 13 disc'sÖn").

  move(H,1,HA,_,HC,CA,_,CH):-!,inform(H,1,HA,HC,CA,CH).
  move(H,N,HA,HB,HC,CA,CB,CC):-
              N1=N-1,
              HA1=HA+1,
              move(H,N1,HA1,HC,HB,CA,CC,CB),
              inform(H,N,HA,HC,CA,CC),
              HC1=HC+1,
              move(H,N1,HB,HA,HC1,CB,CA,CC).

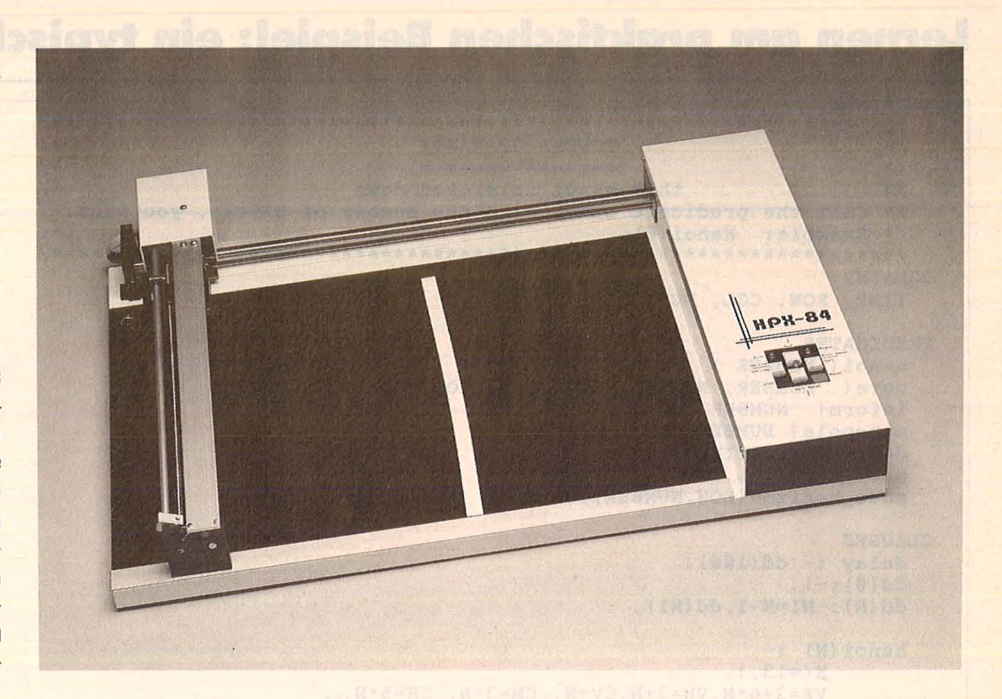
 inform( H, N, H1, H2, C1, C2 ) :-
        C11=C1-N, C22=C2-N, NN=2*N,
        H11=H-H1, H22=H-H2,
        move_vert(C11,NN,H11,1),
        move_horizon(1,NN,C11,C22),
        move_vert(C22,NN,1,H22).

  makepole(_,0,_):-!.
  makepole(H,N,C):-HH=H-N,inform(H,N,HH,HH,C,C), N1=N-1, makepole(H,N1,C).

  move_vert(_,_,H,H):-!.
  move_vert(COL,SIZE,H1,H2):-H1<H2,!, /* move up */
        H11=H1+1,
        field_attr(H11,COL,SIZE,112),
        field_attr(H1,COL,SIZE,7),delay,delay,
        move_vert(COL,SIZE,H11,H2).
  move_vert(COL,SIZE,H1,H2):-H1>H2,!, /* move down */
        H11=H1-1,
        field_attr(H11,COL,SIZE,112),
        field_attr(H1,COL,SIZE,7),delay,delay,
        move_vert(COL,SIZE,H11,H2).

  move_horizon(_,_,H,H):-!.
  move_horizon(ROW,SIZE,C1,C2):-C1<C2,!, /* move right */
        C11=C1+1, HH=C1+SIZE,
        field_attr(ROW,HH,1,112),
        field_attr(ROW,C1,1,7),delay,
        move_horizon(ROW,SIZE,C11,C2).
  move_horizon(ROW,SIZE,C1,C2):-C1>C2,!, /* move left */
        C11=C1-1, HH=C11+SIZE,
        field_attr(ROW,C11,1,112),
        field_attr(ROW,HH,1,7),delay,
        move_horizon(ROW,SIZE,C11,C2).
 goal
```

Türme von Hanoi: Das Spielprogramm gehört zum Turbo-Prolog-System

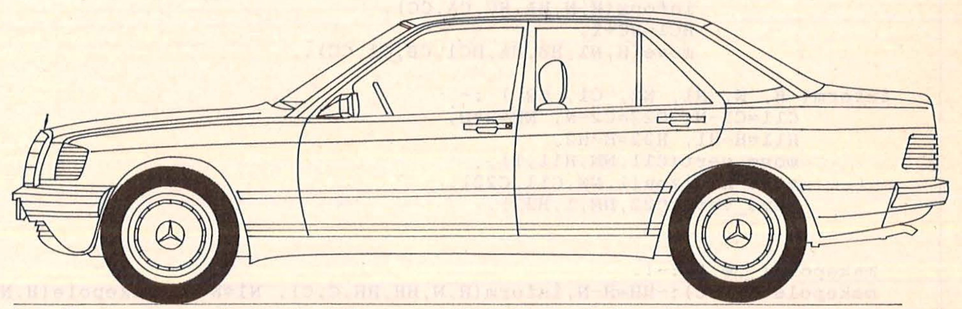


SPLOTTBILLIG

Plotten statt Printen? Der Flachbett-Plotter HPX-84 kostet nicht mehr als ein guter Nadeldrucker

Für Leute, die CAD nicht mehr für ein neues Deospray halten, so ziemlich das Höchste: Ein Plotter als Ausgabegerät, unübertroffen gerade da, wo es auf höchste Präzision ankommt. Aber meist auch entsprechend teuer, da Elektronik, aufwendige Mechanik und kleinere Serie kräftig zu Buche schlagen.

Dem Traum vom elektronischen Zeichenbrett zum Printerpreis kommt der Flachbettplotter HPX-84 von Habersetzer (Polling) schon recht nahe, zumal das Gerät keine Vorurteile kennt: Es arbeitet klaglos mit allen gängigen Rechnern zusammen, die über eine Centronics-Schnittstelle verfügen (Beim Commodore 64 verbindet ein Spezialkabel den Userport mit dem Plotter). Es wäre also eine für CAD-Verhältnisse ungewöhnlich preiswerte Konfiguration denkbar. Der HPX-84 selbst kostet als Fertig-Gerät mit einer Auflösung von 0,1 mm knapp 1400 Mark (0,05 mm Auflösung: 1500 Mark, 0,025 mm Auflösung 1600 Mark). Wer keine Scheu vor dem Zusammenbau der Präzisionsmechanik und rund acht Stunden Zeit hat, kann jedes Modell als Bausatz erwerben und damit jeweils 200 Mark sparen.



Preiswert geplottet: *Mit dem HPX-84 und dem Programm PIC*

Der Plotter bietet eine Arbeitsfläche von 400 × 290 mm, also etwa soviel, wie ein DIN A3-Blatt Platz läßt – zwischen den beiden Magnetstreifen, die das Blatt auf der Fläche fixieren. Den Vorschub auf der X- und Y-Achse übernehmen zwei unipolare Zweiphasen-Schrittmotore, die Kraftübertragung zwischen Motor und Mechanik vermittelt eine Stahlseil-Konstruktion. Unter der Alu-Abdekkung verbirgt sich neben Netzteil und X-Achsen-Schrittmotor die Elektronik, die für die über eine Z 80A-CPU verfügt und die gesamte Mechanik steuert, etwa Beschleunigen und Abbremsen der Motore, das Erzeugen von Kreisen und Rechtecken oder die Randüberwachung. Diverse Stifthalterungen gestatten den Einsatz von verschiedensten Faserschreibern oder Tuschezeichnern. Einfarbig bleibt die Grafik in jedem Fall. Das Design des Plotters wirkt alles andere als preisverdächtig, dafür aber außerordentlich solide und vertrauenserweckend.

Der Plotter reagiert auf die ASCII-Codes, die er vom Computer via Centronics-Schnittstelle empfängt. Sein Zeichenstift bringt also beispielsweise von BASIC aus mit einem PRINT-Befehl den gewünschten Strich aufs Papier. Ein extrem umständliches Vorgehen – wie schon ein Blick in das magere Handbuch zeigt. Im Normalfall wird man zu einem ausgewachsenen CAD-Programm greifen. Für die meisten Anwender dürfte das Programm „Mica" von E & C (Erlangen) interessant sein, das wir in HC 7/86 besprochen haben (Anpassungen an Apple II, Schneider CPC und Joyce, IBM und Kompatible, Commodore 128, Atari ST). Für höhere Ansprüche an IBM & CO gibt es PIC (Pictures by PC), allerdings auch zum zehnfachen Preis.

Im praktischen Einsatz zeigte sich der HPX-84 sehr bedächtig, aber auch präzise: Er produzierte stumm stufenlose Kreise oder Diagonalen und arbeitete mit erstaunlicher Wiederholgenauigkeit (besser als vom Hersteller angekündigt) sowie ohne Ausfallserscheinungen. *-hs*

Bücher zum Commodore 64

VOGEL Computerbücher

Förster, Hans-Peter
Der Heimcomputer als Btx-Terminal
Wie man Btx-Informationen mit dem HC speichert und weiterverarbeitet
196 Seiten, zahlr. Abb., 30,— DM
ISBN 3-8023-0850-6

Bradbury, A.J.
Das Abenteuer-Programmierbuch für den Commodore 64
Erst programmieren — dann spielen
196 Seiten, 18 Abb., 30,— DM
ISBN 3-8023-0809-3

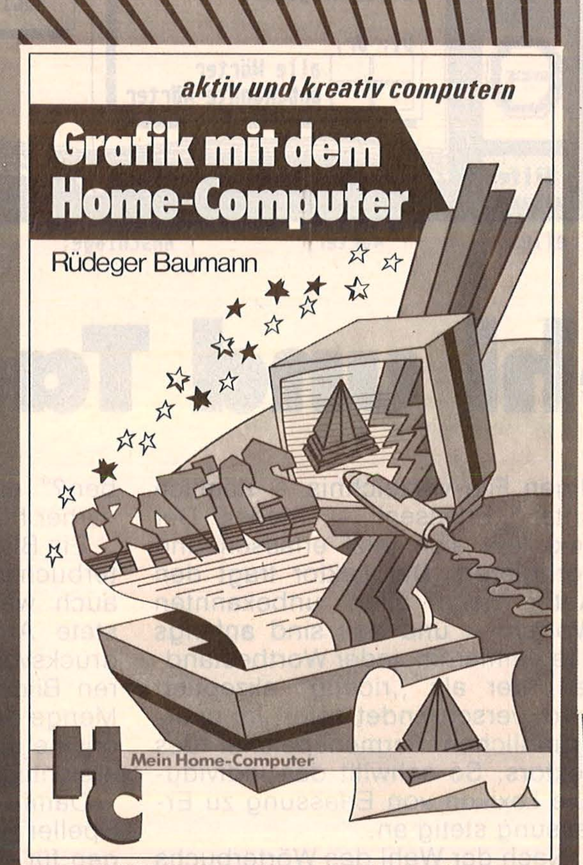
Wittwehr, Clemens
Spiel und Aktion mit Commodore-Logo
Mit der Schildkröte ins Land der Abenteuer
160 Seiten, 42 Abb., 28,— DM
ISBN 3-8023-0851-4

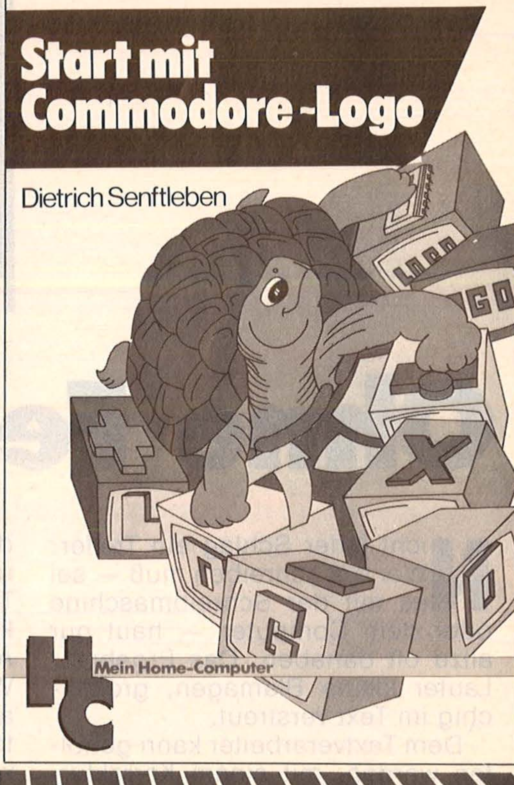
Sinclair, Ian
Mach mehr aus Deinem Commodore 64
Einführung in die Maschinensprache
180 Seiten, 69 Abb., 30,— DM
ISBN 3-8023-0808-5

Kretschmer, Bernd
Multiplan auf dem Commodore 64
Eine systematische Einführung
176 Seiten, 61 Abb., 28,— DM
ISBN 3-8023-0799-2

Rügheimer, Hannes
Spanik, Christian
Mein zweites Commodore-64-Buch
Das Buch das nach dem Handbuch kommt
280 Seiten, 23 Abb., 38,— DM
ISBN 3-8023-0808-5

Honerkamp, Matthias
Jetter, Martin
Fliegen mit dem Mikro
Das Flugsimulationsprogramm "Flight Simulator II" für Apple II, IBM PC und Commodore 64
184 Seiten, 59 Abb., 38,— DM
ISBN 3-8023-0630-9





VOGEL-BUCHVERLAG WÜRZBURG

VOGEL-Computerbücher helfen lernen, verstehen, anwenden

Sie erhalten bei Ihrem Buch- und Computerfachhändler kostenlos das neue Verzeichnis **„VOGEL-Computerbücher '85/86"** mit rund 100 aktuellen Titeln unserer Reihen **CHIP WISSEN** und **HC — Mein Home-Computer.**

Baumann, Rüdeger
Grafik mit dem Home-Computer
Grundlagen und Anwendungen programmiert in BASIC
328 Seiten, zahlreiche Abbildungen

Dieses Buch führt alle Interessierten, die Grundkenntnisse in BASIC haben, in die Prinzipien und Techniken der Grafik-Programmierung auf Mikrocomputern ein. Die Programme sind auf dem Commodore 64 entwickelt und getestet. Sie sind leicht auf andere grafikfähige Mikrocomputer übertragbar.

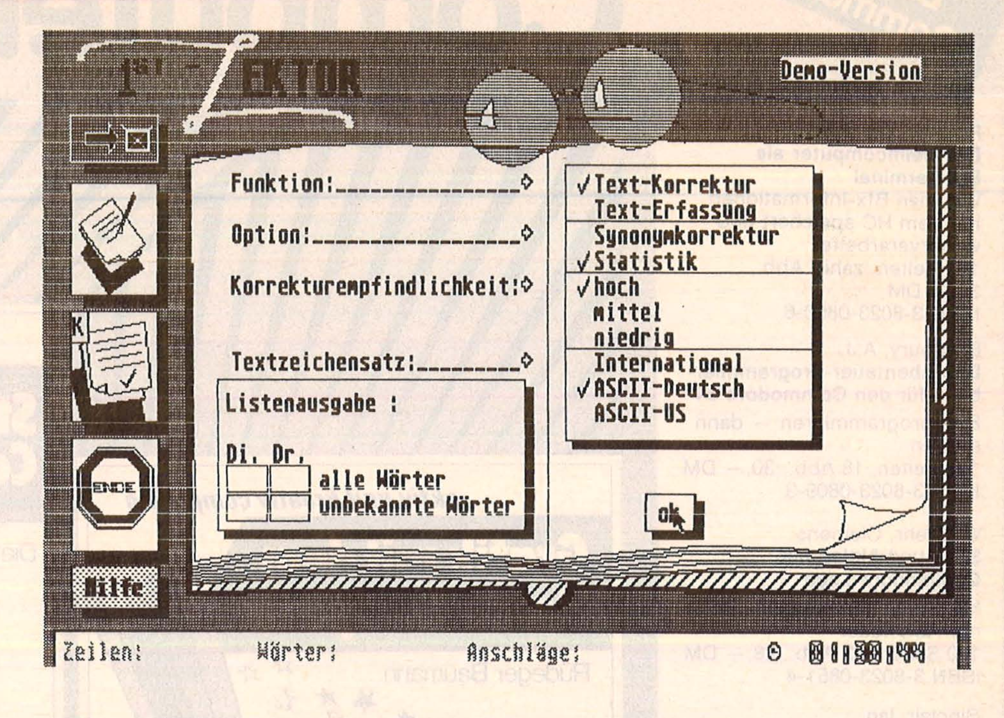
ISBN 3-8023-0769-0 — 38,— DM

Senftleben, Dietrich
Start mit Commodore-Logo
Das kleine Logo-Einmaleins
Grafik · Text · Musik
212 Seiten, 70 Abbildungen

Willkommen bei Logo, der benutzerfreundlichen Computersprache für Freizeit und Ausbildung. In dieser Einführung wird mit Grafik, Text und Musik gespielt, gearbeitet und experimentiert. Das Buch verlangt aktive Mitarbeit und gibt Hilfen und Anregungen für eigenes Forschen.

ISBN 3-8023-0802-6 — 30,— DM

Nieder mit den Tippfehlern, es lebe der perfekte Text: „1st Lektor" bügelt Rechtschreibpannen aus. Auf Atari ST.



Ohne Fehl und Tadel

Nicht jeder Schlag ein Treffer: Wer viel schreiben muß – sei es mit der Schreibmaschine oder dem Computer – haut nur allzu oft daneben. Das Ergebnis: Lauter kleine Blamagen, großflächig im Text verstreut.

Dem Textverarbeiter kann geholfen werden, mit einem Korrekturprogramm, das wenigstens die Orthographie aufpoliert – um Grammatik und Stil muß er sich vorerst noch selber kümmern. Der „1st Lektor" von CCD (Walluf), den wir in der letzten Ausgabe kurz vorstellten, repariert die Rechtschreibdefekte von Atari ST-Besitzern.

Der vorliegenden Version fehlte zwar noch das Lexikon – es soll in der Endfassung etwa 60 000 Wörter umfassen – dennoch hinterließ die zweite Ausgabe einen guten Eindruck, vor allem der höheren Geschwindigkeit wegen, mit der Texte eingelesen und korrigiert werden. Das fehlende Lexikon läßt sich in kurzer Zeit durch die Eingabe eigener Texte ersetzen. Das Programm lernt schnell und willig.

Der Korrekturwillige baggert also erst mal eine Datei von seinem Textprogramm – beispielsweise von „1st Word" – auf den Lektor herüber. Erwartungsgemäß findet sich der Text (als .DOC) im zuständigen File-Verzeichnis – nämlich unter „Erfassen" – wieder. Der Text läßt sich jetzt erfassen und korrigieren. Der Lektor fragt den Autor nach allen unbekannten Worten – und das sind anfangs alle. Immerhin, jeder Wortbestandteil, der als „richtig" akzeptiert wird, verschwindet sofort im unergründlichen Formenspeicher des Lektors. So schwillt das individuelle Lexikon von Erfassung zu Erfassung stetig an.

Nach der Wahl des Wörterbuchs (es lassen sich mehrere anlegen) vergleicht das Programm den Text mit dem eingespeicherten Vokabular und meldet anschließend die Zahl der nicht identifizierbaren Vokabeln. Im linken Fenster erscheint der Text (Im Format von neun Zeilen zu 24 Zeichen), wobei das fragliche Wort besonders hervorgehoben ist. Im rechten Fenster bringt der Lektor Beschwerden („Wort unbekannt"), Fragen („Großschreibung?") oder Verbesserungsvorschläge an, also eine Auswahl an Synonymen, die seiner Meinung nach besser passen könnten. Korrekturen werden sofort auf einer Editierzeile ausgeführt, der bislang unbekannte Wortschatz in das Lexikon abgespeichert. „Soll ich die neuen Wörter in ein Wörterbuch schreiben?" erkundigt sich der Lektor vorher höflich.

Ein Blick in das verborgene Wörterbuch ist jederzeit möglich, wenn auch wenig aufschlußreich. Das stete Anwachsen registriert eindrucksvoller ein Zählwerk am unteren Bildschirmrand, wo auch die Menge der Unbekannten, der Synonyme und die aktuelle Uhrzeit (!) ersichtlich sind.

Damit wäre für einen normalen Speller Feierabend. Nicht dagegen für den „1st Lektor". Der legt jetzt erst richtig los und bombardiert den Autor mit Statistiken sowie wahlweise Balken-, Linien- oder kombinierten Grafiken. Von A bis Z erfährt man grafisch und numerisch

★ den Anteil jedes einzelnen Wortes im Gesamttext (in Prozenten)
★ die Zusammenfassung der zehn größten Werte
★ die Anzahl der einzelnen Wörter
★ die Zusammenfassung der zehn häufigst gebrauchten Vokabeln
★ die Häufigkeit der einzelnen Anfangsbuchstaben
★ Die Häufigkeit jedes Buchstabens insgesamt

So bietet das Programm für 150 Mark zwar zusätzlichen Unterhaltungswert, aber auch überflüssigen Ballast.

-hs

Geschwister Wirbelwind

Die eigentliche Arbeit beim Commodore Amiga erledigen drei Chips mit den klangvollen Namen Paula, Agnus und Denise

Um den Leser vorneweg ein wenig zu verwirren, sei verraten, daß die drei Koprozessoren Paula, Agnus und Denise eigentlich ein Superchip sind, in dem fünf Prozessoren untergebracht sind. Zur Auflösung des Wirrwarrs: Koprozessoren sind bekanntlich Prozessoren, die unabhängig vom Hauptprozessor der Arbeit nachgehen, für die sie speziell entwikkelt sind.

In der Kooperation zwischen CPU und Koprozessor tritt bei den meisten Computern das Problem auf, daß nur die CPU auf den RAM-Speicher zugreifen kann. Reichen bei umfangreichen Aufgaben die internen Register des Koprozessors nicht aus, muß er die CPU zum Durchschalten ins RAM mißbrauchen.

Weil es aber Bitsalat gibt, wenn CPU und Koprozessoren gleichzeitig auf den Speicher zugreifen, ist der RAM-Speicher mit 14,4 Megahertz doppelt so schnell getaktet wie die CPU (7,2 MHz). Nur jeder zweite Takt des RAM-Speichers wird dabei für die CPU reserviert, die anderen werden auf die Koprozessoren verteilt.

Paula, die Sängerin

Der Name ,,Paula" leitet sich irgendwie – auch bei Commodore ist das nicht mehr nachzuvollziehen – von ,,Peripheral/Audio" ab. Denn dieser Prozessor kontrolliert die Arbeit des Computers mit der Peripherie und ist für die Klangerzeugung zuständig. Daneben ist Paula verantwortlich für das gesamte Interrupt-Handling.

Bei einem Interrupt wird das gerade laufende Programm angehalten, sein momentaner Zustand gesichert und eine spezielle Routine gestartet, die diese Unterbrechung bearbeitet. Paula verbindet die Interrupts der Ein-/Ausgabe-Bausteine, der anderen Koprozessoren und des Floppycontrollers und gibt sie als entsprechende Signale an die Register, in denen die CPU nachsieht, was sie bei einer Unterbrechung tun soll. Da Paula aber auch Maus, serielle Schnittstelle, Drucker, Floppy, Harddisk und alle anderen angeschlossenen Geräte verwaltet, werden zum Beispiel Interrupts, die der Drucker schickt (,,Daten gedruckt, bitte weitere Daten schicken") direkt, ohne den Hauptprozessor zu behelligen, verarbeitet, indem Paula via Agnus dem Drucker eigenständig weitere Daten aus dem RAM-Speicher bereitstellt. Die Ausgabe eines Textes beim Amiga auf den Drucker kann also erfolgen, während der Benutzer gleichzeitig auf dem Computer etwas ganz anderes tut.

Darüber hinaus besitzt der Paula-Chip vier Soundkanäle, von denen je zwei als Stereokanäle zusammengefaßt werden können. Er kann analoge Signale (Schallwellen) in digitale Impulse umsetzen und umgekehrt. Um eine Melodie parallel zu einem anderen Programm spielen zu lassen, holt Paula mit DMA die zur Musikerzeugung nötigen Daten eigenständig aus dem Speicher und gibt sie als analoge Wellen auf angeschlossene Lautsprecher aus, indem sie die Spannung, die am Lautsprecherausgang anliegt, anhand der gespeicherten Daten verändert.

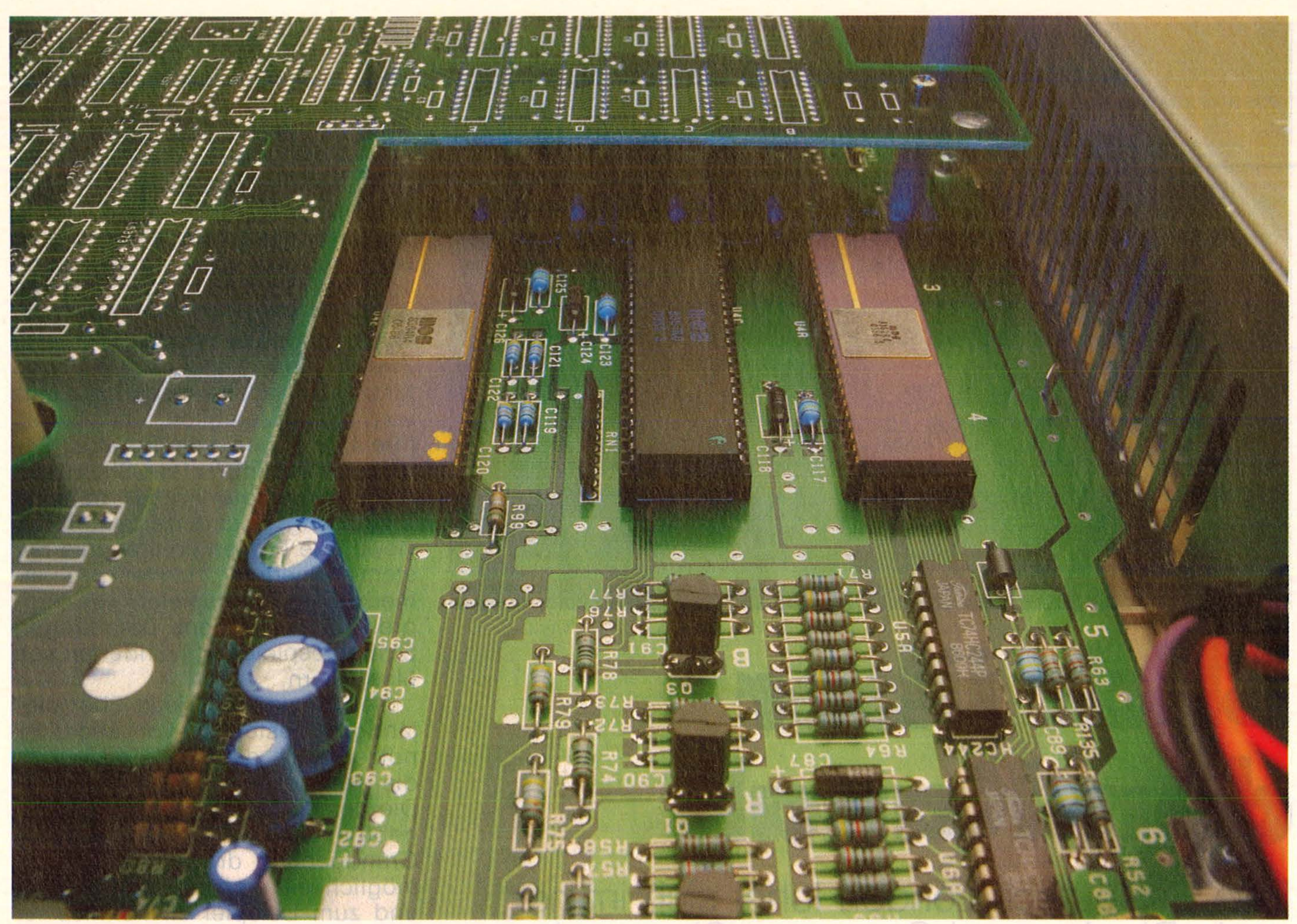
Der Name ,,Denise" kommt irgendwie von ,,Display-Encoder" (Bildkodierer). Sobald der Amiga irgendetwas grafisches tun muß, kommt Denise (zusammen mit Agnus) ins Spiel. Denise besitzt 32 Farbregister, in denen jeweils die Rot-, Grün- und Blauintensität der entsprechenden Farbe gespeichert werden kann. Insgesamt sind bekanntlich 4096 Farbtöne möglich. An Sprites kann Denise acht verschiedene gleichzeitig auf dem Bildschirm bewegen, jeden 16 Pixel breit und beliebig hoch. Jeder dieser acht Sprites kann – wie beim VIC des C64 mehrmals verwendet werden, so daß es nur von der Geschicklichkeit des Programmierers abhängt, wieviele Sprites letztendlich tatsächlich auf dem Bildschirm zu sehen sind. Um dem Amiga die kommerzielle CAD-Welt zu öffnen, in der eine Minimalauflösung von 1024 x 1024 Bildschirmpunkten verlangt wird, kann Denise auch eine größere Grafikfläche als den Bildschirm verwalten.

Denise, die Malerin

Dieses sogenannte ,,Playfield" kann stufenlos hin- und hergescrollt werden, so daß auf dem Bildschirm immer nur ein Ausschnitt der tatsächlich vorhandenen Grafik zu sehen ist. Ein Wunderchip ist Denise also nicht, eher ein aufgemotzter VIC mit dessen bekannt guten Grafikeigenschaften.

Von seiner Funktion als Adreßgenerator kommt irgendwie der Name ,,Agnus". Agnus beinhaltet alle DMA-Kanäle, verwaltet sie und ermöglicht damit den anderen Koprozessoren den Zugriff auf den RAM-Speicher. Agnus wartet solange, bis die CPU den Speicher nicht mehr benötigt und schlägt dann zu: Solange ein DMA-Zugriff über einen der Koprozessoren erfolgt, signalisiert Agnus der CPU, sich vom Speicher fernzuhalten. Erst danach gibt er ihn wieder frei, was zum Beispiel bei umfangreichen Grafikaufgaben dazuführt, daß der gesamte Rechner ,,in die Knie" geht. Damit wird verhindert, daß zwei oder mehrere Prozessoren gleichzeitig auf den RAM-Speicher zugreifen, was heilloses Chaos verursachen würde.

Im Agnus-Chip sind zwei weitere Prozessoren integriert: Der Blitter



Die glorreichen Drei: *Paula, Denise und Agnus animieren den Amiga zu Höchstleistungen*

und der Copper. Der Blitter schaufelt im wesentlichen große Datenmengen schnell durch den Speicher. Er ist verantwortlich für das fixe Linienziehen und Flächenausfüllen und für die BOB's (Blitter Objects), die so etwas ähnliches sind wie sehr große Sprites. Bei den BOB's werden die Bitmaps, die ihr Aussehen bestimmen, in den Adreßraum des Bildschirmspeichers hineinkopiert, auf dem das BOB zu sehen ist. Gleichzeitig wird der alte Inhalt des Bildschirmspeichers in einen Puffer kopiert. Wird das BOB bewegt, wird die BOB-Bitmap an die neuen Bildschirmadressen und der Pufferinhalt an seine alte Position zurückkopiert. Das ist wichtig bei den Fensterbewegungen und den verschiedenen überlappenden virtuellen Bildschirmen, die der Amiga besitzt. Bei dem Schließen eines Fensters muß das dahinterliegende nicht mehr neu aufgebaut werden, sondern lediglich vom Blitter der Inhalt des Puffers, in dem das Aussehen des verdeckten Teils des Fensters liegt, an die alte Stelle des Bildschirms zurückkopiert werden, was natürlich wesentlich schneller geht. BOB's sind damit als grafische Elemente wesentlich vielseitiger als Sprites, die ganze Kopiererei im Speicher ko-

Agnus, der Verwaltungschef

stet aber mehr Rechenzeit, als nur wenige Spritepixel vor dem (unverändert bleibenden) Hintergrund zu bewegen, so daß Sprites als bewegliche Grafik wesentlich sinnvoller weil schneller sind.

Der Copper (von: Co-Prozessor) schließlich ist eng an den Elektronenstrahl gekoppelt, mit dem der Monitor sein Bild aufbaut. Dieser Prozessor hat die Fähigkeit, zu erkennen, in welcher Zeile und Spalte des Bildschirms der Kathodenstrahl gerade ist. Bei entsprechender Programmierung ist es dem Copper möglich, sobald ein bestimmter Punkt erreicht ist, die Register anderer Chips zu verändern. Was unter Umgehung des Hold-and-Modify-Modus dazu verwendet werden kann, sämtliche 4096 Farben auf dem Bildschirm darzustellen, indem der Copper einfach in den Farbregistern der anderen Chips herumpfuscht. Auch wenn der Benutzer mit seiner Maus einen virtuellen Bildschirm herunterzieht, um nachzusehen, was seine anderen Programme gerade machen, sorgt der Copper dafür, daß jedes Programm weiß, wo der Bildschirm anfängt, auf den es malen darf. Ausgefuchste Hardware-Bastler, die nun Blut geleckt haben und in ihren selbstgebauten Rechner den einen oder anderen dieser Superchips einbauen wollen, müssen enttäuscht werden: Im Gegensatz zum Motorola 68000, dem Hauptprozessor des Amiga, sind Agnus, Denise und Paula „Custom-Chips", die in keinem Laden und bei keinem Dealer einzeln zu kaufen sind. *Joachim Graf*



Blick über den Kanal

Eine Reise nach London lohnt sich immer noch. Die Amstrad-Consumer-Show gab Aufschluß über künftige Entwicklungen nicht nur bei Schneider-Computern.

Nichts zu sehen und nichts zu hören gab es über den IBM-kompatiblen PC von Amstrad, am besten sollte man also auch nicht drüber sprechen. Während hierzulande die Gerüchteküche überbrodelt (neuester Stand: Vorstellung eines IBM-kompatiblen Schneider-PC auf der Orgatec in Köln im Herbst), gab sich Mike Mordecai, Software Sales Manager von Amstrad, bedeckt. Gute Tradition von Amstrad sei es, daß nichts nach außen dringt, bevor ein Produkt perfekt ist.

Die gleiche Seriösität legte er jedoch nicht an den Tag, als es um den kürzlich von Sinclair übernommenen Home-Computer-Bereich ging. Ein neuer Spectrum mit 128 KByte RAM, wie der Spectrum plus, jedoch mit eingebautem Kassettenrecorder und eingebautem Joystick, sei für den Herbst denkbar. Schneider hatte bereits vor diesen Plänen den Schritt von Amstrad (Übernahme von Sinclair) nicht mitgemacht. Daß am neuen Spectrum was dran sein könnte, zeigt die Auskunft von Jürgen Schumpich, ehemaliger deutscher Generalimporteur von Sinclair-Produkten, daß er mit Amstrad bereits Verhandlungen aufgenommen habe.

In einer Hinsicht ist die Bundesrepublik Deutschland sicherlich noch Entwicklungsland: bei der Datenfernübertragung. Dabei sind nicht nur die Vereinigten Staaten als Vorbild zu nennen, auch unsere Nachbarländer sind uns deutlich voraus. Insbesondere Frankreich: weit über 1,5 Millionen Bürger sind dort an das Télétel-Netz angeschlossen (vergleichbar mit Btx). Die Amstrad-Show in London zeigte, daß auch in Großbritannien sich wesentlich mehr tut, wenn es gilt, an einer Zukunft zu bauen, in der zum Beispiel Datenbankbenutzung und Kommunikation über Mailboxen zu den Selbstverständlichkeiten des Alltags gehören.

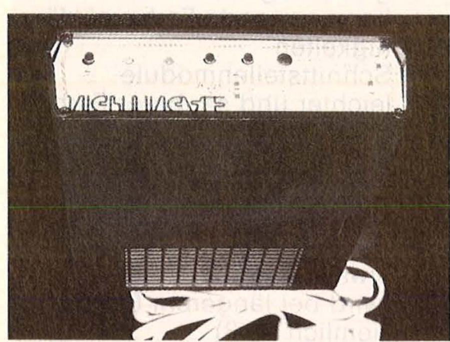
Die Voraussetzungen dazu sind allerdings nach der Privatisierung von British Telecom gänzlich anders als in Frankreich oder der Bundesrepublik Deutschland. Zwei Systeme, micronet 800 und Microlink, haben sich auf der Basis von Prestel etabliert und bauen ihre Leistungen ständig aus. Genannt seien nur: Tele-Shopping, Sonderangebote in der näheren Umgebung, Kleinanzeigen, Interviews (die Fragen können alle System-Teilnehmer stellen), Kummerkasten, Überspielen von Programmen auf eigene Datenträger, Strategiespiele mit Hunderten von Mitspielern.

Seit Beginn des jetzt über einjährigen Bestehens von Microlink nutzen auch deutsche Teilnehmer die Möglichkeiten des Systems. Die Kosten für Microlink: Beitrittsgebühr fünf Pfund, monatliche Grundgebühr drei Pfund, Telexmöglichkeit einmalig zehn Pfund und zum Beispiel elf Pennies für 100 Zeichen innerhalb Europas oder 2,75 Pfund für 400 Zeichen zu Schiffen auf See. Microlink wird von Database Publications in Hazel Grove betrieben. Um den Markt richtig erschließen zu können, bringt Database Publications die DFÜ-Zeitschrift „Telelink" heraus. Eigentlich sollte es nur eine Frage von kürzester Zeit sein, bis man auch in der Bundesrepublik Deutschland mit einer DFÜ-Zeitschrift einer sich abzeichnenden Entwicklung Rechnung trägt.

Am Stand von Northern Computers aus Frodsham war eine Festplatte für den Amstrad mit einer Kapazität von 20 Megabyte zu sehen. Das entspricht ungefähr dem Hundertfachen von dem, was man auf einer Diskette unterbringen kann. Besonders interessant ist die Festplatte, wenn mit einem Netzwerk von der gleichen Firma bis zu 100 Amstrad-Computer auf die gleichen Programme zugreifen können. dk'tronics aus Great Yarmouth hatte sich bisher stark für Sinclair engagiert und bietet jetzt für Amstrad zahlreiche Erweiterungen an: eine 64K-Speichererweite-

rung, eine 256K-Speichererweiterung, eine Speichererweiterung, die 256K als RAM-Disk zur Verfügung stellt, einen Sprach-Synthesizer und ein Grafikpaket. Alle fünf können als Steckmodule gleichzeitig benutzt werden.

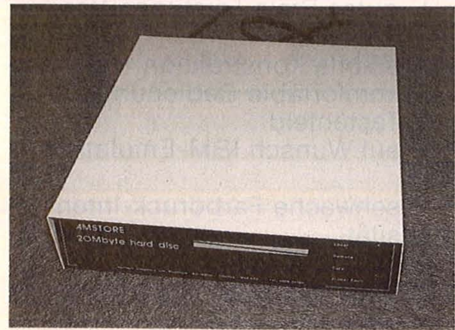
Ein Midi-System von EMR in Wickford, das schon für BBC, Commodore und Spectrum angeboten worden war, steht jetzt auch für den Amstrad zur Verfügung. Dem Preis von 80 englischen Pfund für das Midi-Interface muß man den Preis für die Grund-Software hinzurechnen, ohne die man nicht vernünftig arbeiten kann. Das System bietet unter anderem acht Stimmen polyphon, 16 Midi-Kanäle, neun Oktaven und eine Kapazität von bis zu 8000 Noten. Software-Hilfen zum Komponieren, Erzeugen neuer Klänge, Schreiben von Partituren, eine Datenbank mit einer Auswahl von Klängen, grafische Interpretation von Musik werden das Midi-System bald abrunden.



DFÜ: *Für Modem-Benutzer bietet Großbritannien einiges*



Erweiterungen: *Von RAM-Disk bis Sprachsynthesizer*



Harddisk: *Die hundertfache Kapazität einer Diskette*



Midi: *An Musikalität steht Amstrad anderen nicht mehr nach*

Für Schneider-Benutzer, die innerhalb der Rechner-Familie von einer Kassettenversion auf eine Diskettenversion umgestiegen sind, leisten Transfer-Programme einen nützlichen Dienst. Siren Software in Manchester bietet ein solches an (acht Pfund auf Kassette, elf Pfund auf Diskette), gesteht aber ein, daß die Übertragung nur bei 80 Prozent aller Programme erfolgreich ist. Eine Hardware-Lösung für dieses Problem kommt von Mirage in Braintree. Auf Knopfdruck an einem Steckmodul soll jedes Programm automatisch übertragen werden können, und zwar nicht nur von Kassette auf Diskette, sondern auch umgekehrt.

Zusammen mit einer Maus bietet Advanced Memory Systems Programme an, die eine Benutzeroberfläche schaffen, die stark an Apple Macintosh, Atari ST oder Amiga erinnert. Es können Fenster benutzt und in eigenen Programmen sinnvoll eingesetzt werden, Icons sind installiert oder können selbst entworfen werden. Ein solides Grafiktablett kommt von Grafsales in Watford. Die verbesserte Version ist geeignet für Heim- und Geschäftsanwendungen. Das Einsatzgebiet könnte jedoch vor allem die Erzeugung von Spiele-Grafik und die Weiterbildung sein.

Kräftige Unterstützung fand auf der Amstrad-Show der Joyce (in Großbritannien heißt er PCW8256 bzw. 8512). In Vorstellung und Beratung gab man sich ganz professionell. In einem ständig überfüllten eigenen Raum wurde über Projektionsschirm in die Software für den Joyce eingeführt. Eintageskurse für die Logoscript-Textverarbeitung auf dem Joyce werden in verschiedenen Städten Großbritanniens abgehalten. Wer von dem Textverarbeitungssystem Tasword auch auf dem Joyce nicht lassen möchte, weil er es von anderen Computern her gewohnt ist, kann für 25 Pfund Tasword 8000 kaufen.

Eine ganze Reihe von professioneller Software für den Joyce wird von den Firmen Caxton Software in London, Saxon Computing in Leconfield, Saga Popular Software in Newcastle, Gemini in Exmouth, Compsoft, Cavalier Software, Pos Supplies und Sifan Marketing (alle in London) angeboten. Die Palette reicht von Kalkulation, Dateiverwaltung über Sprachen, Kurse, bis zu DFÜ-Programmen.

Bei der Lern-Software tut sich nicht allzuviel. Einige Buchstaben- und Zahlenprogramme für Schulanfänger sowie Mathematik- und Fremdsprachenprogramme nehmen nicht den großen Anteil am gesamten Software-Angebot ein, den man der Lern-Software noch vor kurzem prophezeit hat.

Eingefleischte Spiele-Freaks sind auch auf dieser Amstrad-Show wieder gut bedient worden. Für einen eher Außenstehenden fällt höchstens auf, daß das Faible der Engländer für alle Ballspiele auf sattem grünen Rasen mit Fußball, Tennis, Cricket, Golf und Polo auch auf dem Bildschirm zum eher matten Ausdruck kommt. Brutale Kämpfe, Flugzeug gegen Flugzeug in der Luft und Mann gegen Mann auf der Erde (vorwiegend mit ostasiatischem Einschlag) scheinen nach wie vor angesagt zu sein. Zuständig für den Spielebereich fühlen sich vor allem Mirrorsoft in Bristol, der Home-Computer-Club in Swindon und Audiogenic Software in Reading.

Mit Literatur ist der Amstrad in Großbritannien reichlich versorgt. In den Büchern wird mehr und mehr fundiertes Wissen über Programmiersprachen und Betriebssysteme (insbesondere CP/M plus) vermittelt. *-br*



Farbdrucker

Okimate 20



Der Okimate-Farbdrucker ist zwar der billigste Drucker in unserer Gegenüberstellung, gehört jedoch von seinen Leistungsdaten her mit zu den Besten. Im Gegensatz zu seinen Druckerkollegen arbeitet der Okimate mit einem Thermofarbband (farbig oder schwarzweiß) und erreicht damit leuchtende und kontrastreiche Farbdrucke in bis zu 14 Farben. Entsprechend angenehm ist sein Arbeitsgeräusch: Außer dem leisen Surren der beiden Elektromotoren für den Druckkopf- und Walzenantrieb ist nichts zu hören.

Und auch an Universalität kann es der kleine Drucker durchaus mit seinen professionelleren Kollegen aufnehmen. Dazu bietet Okidata für die bekanntesten Computermodelle verschiedene Schnittstellenmodule (V24, Centronics, C64-IEC) an, welche sich leicht in eine entsprechende Öffnung an der Seite des Okimate einstecken lassen. Der große Vorteil dieser genialen Idee ist folgender: Man erhält nicht nur einen auf den jeweiligen Rechner optimal angepaßten Drucker, sondern kann ihn bei einem späteren Systemwechsel leicht und preisgünstig an seinen neuen Computer anpassen. Doch das ist immer noch nicht alles: Okidata hat sich für Commodore-Besitzer einen besonderen Service einfallen lassen. Für den C64 ist ein spezielles Print-Set erhältlich, welches neben umfangreichem Zubehör (Farbbänder, Thermopapier usw.) eine randvoll gepackte Diskette mit phantastischen Demobildern und guten Ausdruckprogrammen für alle gängigen Malprogramme (Doodle, Paint Magic, Koala Painter usw.) enthält.

VOR- UND NACHTEILE

+ günstiger Preis
+ gutes Begleitmaterial
+ hervorragende Farbgrafikfähigkeiten
+ Schnittstellenmodule
+ leichter und sauberer Farbbandwechsel

– ziemlich teure Farbbänder
– schwieriger Einzelblatteinzug
– etwas schwaches Netzteil (wird bei längerem Betrieb ziemlich heiß)

Epson EX-800



Jahrelange Erfahrung in der Entwicklung und Produktion von Druckern hat sich in diesem Gerät niedergeschlagen und macht es gleichermaßen für den anspruchsvollen Büro- wie dem ambitionierten Hobbyeinsatz zum idealen Drucker. Ein Blick ins Innere zeigt einen stabilen Druckkopf mit wuchtigem Kühlkörper (und aufliegender Farbbandkassette) für den harten Dauereinsatz. Der anfällige Keilriemen zum Transport des Druckkopfes wurde gegen ein stabiles Stahlseil ausgetauscht. Auf der Rückseite befindet sich neben den Schnittstellenbuchsen (Centronics, V24 optional) und den frei zugänglichen DIP-Schaltern ein großzügig bemessener Kühlkörper für die Steuerelektronik.

Bereits vom FX-85 ist die Auswahl der Druckmodi über das Tastenfeld bekannt. Es stehen jetzt insgesamt acht beleuchtete Auswahltasten zur Verfügung, die das schnelle Umschalten der Schriftarten und Optionen zum reinen Vergnügen machen.

Natürlich ist der EX-800 voll kompatibel zu seinen beiden legendären Vorgängern FX-80 und JX-80 und ermöglicht neben NLQ-Druck auch die IBM-Emulation.

Bei soviel Sonne muß es zwangsläufig auch etwas Schatten geben: Mal davon abgesehen, daß sich im Hinblick auf die mangelnde Farbintensität des JX-80 nicht viel geändert hat, ist der Lärmpegel während des Druckens nur vom Nebenzimmer einigermaßen erträglich (Kreissägen-Syndrom!). Trotzdem bleibt der EX-800 sehr empfehlenswert.

VOR- UND NACHTEILE

+ gutes Preis-Leistungs-Verhältnis
+ stabile Konstruktion
+ komfortable Bedienung über Tastenfeld
+ auf Wunsch IBM-Emulation

– schwache Farbdruck-Intensität
– mäßige Farbband-Lebensdauer (30% der Lebensdauer des einfarbigen Farbbandes)

im Vergleich



Juki 5520

Mit seinem neuen Modell 5520 stellt Juki einen Farbdrucker für den Hobby- und Bürobereich vor. Im Gegensatz zum Okimate arbeitet der Juki-Drucker mit einem mehrfarbigen Textilband. Entsprechend lauter ist auch der Druckvorgang. Wirklich vorbildlich ist das Handbuch gelungen, was neben der Beschreibung der Druckerbedienung ebenfalls hilfreiche Tips für eine Vielzahl von Computern liefert. Oberstes Gebot bei den Juki-Technikern war offensichtlich die Einhaltung der nahezu vollständigen Kompatibilität zum Epson-Standard (FX-80, JX-80). Nicht nur die Escape-Sequenzen zur Steuerung der Druckerfähigkeiten, sondern auch die Anordnung der DIP-Schalter sind identisch zum großen Original.

Ebenfalls sind die gängigen Standardschnittstellen (Centronics, V24 als Option) erhältlich. Leider gibt es für den Commodore 64 kein eigenes Modul, doch kann man sich hier an dem großen Angebot Epson-kompatibler Schnittstellen (wie Görlitz, Wiesemann usw.) bedienen. Ebenfalls erwähnenswert ist, daß der Juki-Drucker per DIP-Schalter auch mit IBM-Druckern kompatibel wird. Damit mausert sich dieses Gerät zum Allround-Talent. Und wer sich einen der vielen billigen IBM-Kompatiblen auf seinen Arbeitstisch stellen will, ist mit dem Juki sicherlich gut beraten.

Wer anfangs vielleicht aus finanziellen Gründen auf die Farbdrucke aus dem Juki 5520 verzichten muß, sollte einmal den kleineren Bruder 5510 anschauen. Er ist in der Grundausstattung als Schwarzweiß-Drucker erhältlich, läßt sich später auf Farbe und volle Epson/IBM-Verträglichkeit umrüsten.

VOR- UND NACHTEILE

\+ volle Epson/IBM-Kompatibilität
\+ Nachrüstung des Juki 5510 möglich
\+ Centronics/V24-Schnittstelle vorhanden
\+ gute Dokumentation mit vielen Hilfestellungen

– langsamer NLQ-Druck
– Farbdruck-Qualität stark abhängig vom Alter des Farbbandes



Comko Farbdrucker

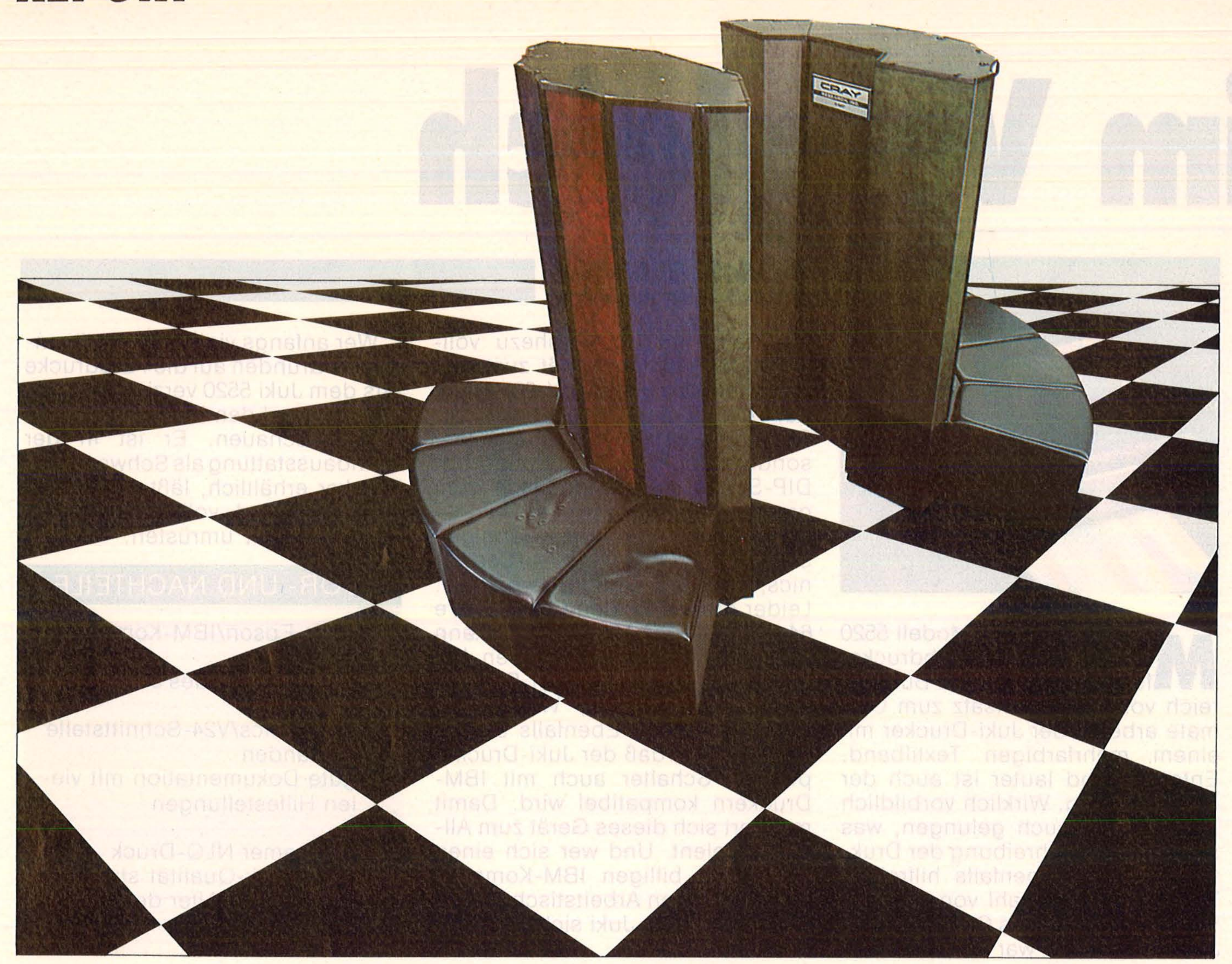
Für den hauptsächlichen Einsatz in Büros ist der Comko Farbmatrixdrucker gedacht. Entsprechend solide ist seine Mechanik ausgefallen, die selbst die des EX-800 in den Schatten stellt und mit über 20 Kilogramm Gewicht kaum für häufigen Transport geeignet ist. Zwei starke Ventilatoren sorgen für Kühlung und schalten sich in den Druckpausen eigenständig ab. Auffallend ist das Bedienungsfeld mit zwölf Funktionstasten, welches wie beim EX-800 die Auswahl von Zeichensätzen und Druckfunktionen ermöglicht. Die gerade ausgewählte Funktion wird dabei über eine rote Digitalanzeige gemeldet.

Auch Anschlußprobleme dürften der Vergangenheit angehören, da der Drucker serienmäßig mit Centronics- und V24-Schnittstelle ausgeliefert wird. Typisch für einen Bürodrucker dieser Leistungsklasse ist der große Vorrat an auswählbaren Zeichensätzen, welcher sich durch steckbare ROM-Module erweitern läßt. Wem das noch nicht reichen sollte, der kann den eingebauten Datenpuffer als Speicher für selbsterstellte Zeichensätze benutzen. So etwas bietet zwar auch der EX-800, doch kann man beim Comko-Drucker aufgrund seiner hervorragenden Druckauflösung Hilfsprogramme benutzen, um digitalisierte Zeichensätze aus Buchdrucker-Vorlagen einzusetzen. Übrigens beträgt die Druckauflösung maximal 240 × 480 Punkte pro Zoll und erreicht damit Laserdruck-Qualität (Standard 300 × 300 Punkte pro Zoll). Erfreulich leise gibt sich das Gerät während des Druckvorgangs, was wohl an dem mit Schaumstoff ausgepolsterten Gehäuse liegen mag.

VOR- UND NACHTEILE

\+ ausgezeichnete Druckauflösung
\+ kontrastreiche Farbdrucke
\+ hervorragender Bedienungskomfort
\+ Typenrad-kompatibel
\+ gute Erweiterungsmöglichkeiten

– hoher Preis
– großes Gehäuse



David gegen Goliath

Mitte Juni dieses Jahres wäre in Köln beinahe ein 8-bit-Rechner Weltmeister im Computer-Schach geworden und hätte den Superrechner Cray Blitz geschlagen

Das erste lauffähige Schachprogramm entstand Mitte der fünfziger Jahre – in der Wüste von New-Mexico. In Los Alamos hatten die Vereinigten Staaten ein riesiges Laboratorium errichtet, um die Entwicklung von Kernwaffen voranzutreiben. Zur Berechnung der Sprengladungen, die die atomare Kettenreaktion zünden sollten, benutzte man einen neuentwickelten Rechner, den Maniac I, der in der Lage war, über 10 000 Berechnungen pro Sekunde auszuführen. Noch wichtiger: er war frei programmierbar.

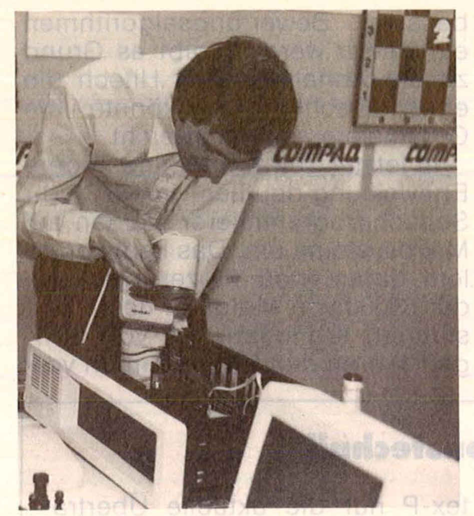
Der neue Computer, ein röhrenbestücktes Monstrum, reizte zum Experimentieren, und es dauerte nicht allzu lange, bis das erste Schachprogramm fertig war. Insgesamt fünf der anwesenden Wissenschaftler beteiligten sich an diesem Projekt. Um die Aufgabe zu vereinfachen, beschlossen sie, das Spielfeld auf ein 6 × 6 Brett zu begrenzen und die beiden Läufer sowie zwei Bauern wegzulassen. Ansonsten blieben fast alle Schachregeln unverändert.

Das Programm wurde Mitte der fünfziger Jahre fertiggestellt und spielte insgesamt drei Partien. Die erste war gegen sich selbst (Weiß gewann), die zweite gegen einen starken Spieler, der dem Computer eine Dame vorgab. Diese Partie dauerte zehn Stunden und endete mit einem Sieg für den Schachmeister. Die dritte Partie spielte eine junge Dame, die eine Woche zuvor die Schachregeln gelernt hatte. Sie verlor das Spiel gegen den Computer. Es war ein historischer Augenblick, denn zum ersten Mal in der Geschichte wurde ein Mensch im Schachspiel von einer Maschine besiegt.

Warum wird man Schachprogrammierer?

Seit jenen Anfängen ließ das Interesse für die Schachprogrammierung keineswegs nach. An den verschiedensten Universitäten und Rechenzentren bildeten sich Arbeitsgruppen, die intelligentere Schachalgorithmen und schnellere Suchstrategien ersannen. Es gab aber auch verbissene Einzelgänger, die Hunderte von Stunden vor einem Terminal verbrachten und neue Ideen für ein stärkeres Programm ausprobierten. Heute existieren mehrere Dutzend ernstzunehmende Fachgruppen in aller Welt, vor allem in den Vereinigten Staaten und Großbritannien, aber auch in der Bundesrepublik Deutschland, Österreich, Holland, in den skandinavischen Ländern, ja sogar in der UdSSR, Ungarn und Rumänien, wo Rechnerzeit gewiß nicht leichtfertig vergeben wird. Alljährlich werden Computerschachkonferenzen und Turniere abgehalten, zu denen die Spitzenprogrammierer um den halben Erdball reisen.

Es gibt viele Gründe, weshalb die Schachprogrammierung diese immense Popularität erlangt hat. Einerseits ist die Aufgabe klar definiert, die Regeln des Spiels sind vollständig bekannt, und das Endziel (das Schachmatt) ist einfach zu beschreiben. Zudem ist das Problem weder zu einfach (wie etwa Dame oder Backgammon, bei denen Computer problemlos zur Meisterstärke gelangen) noch zu kompliziert (wie das japanische Brettspiel Go, bei dem sie kaum das Niveau eines blutigen Anfängers erreichen können). Hinzu kommt, daß der Schachprogrammierer die Fortschritte seiner Arbeit besonders deutlich erkennen und messen kann: Schachpartien lassen sich vorzüglich aufzeichnen, nachspielen und analysieren. Man kann sie nachträglich nach Fehlern absuchen, sie mit Kollegen besprechen und sich von erfahrenen Schachspielern beraten lassen. Und die alljährlichen Turniere zeigen immer aufs deutlichste, ob man im Vergleich zu den Kollegen Fortschritte gemacht hat oder nicht. „Mein Programm ist besser als dein Programm" ist in der Schachprogrammierung eine klare, nachprüfbare Aussage. Schließlich ist Schach ein Kampfsport, und es gibt für den ehrgeizigen Programmierer kaum etwas aufregenderes (und meist auch nervenaufreibenderes), als die Partien seines Computers beim Turnier zu verfolgen. Die Spielleidenschaft der Menschen kommt hier voll zur Geltung.



Ein Handventilator *wurde zur Kühlung der 6502-Hardware des Rebel-Programmes eingesetzt*



Demonstrationsbrett: *Fachleute kommentieren den Spielverlauf*

Computerschach-Weltmeisterschaften

Die erste Computerschach-Weltmeisterschaft fand 1974 in Stockholm statt, mit 13 Teilnehmern aus USA, Kanada, England, Österreich, Ungarn und der UdSSR. Das russische Programm Kaissa wurde mit 3,5 Punkten aus vier Partien erster Weltmeister der Geschichte. 1977 traf man sich zur zweiten Computerschach-WM in Toronto, und diesmal siegte das amerikanische Programm Chess 4.5. Auch in den Jahren danach beherrschte dieses Programm die Szene: Es gewann mehrere nationale Computer-Turniere und nahm auch mit beachtlichem Erfolg an menschlichen Turnieren teil. Die Vorherrschaft der Chess-Serie wurde durch die Verwendung eines Supercomputers (Cyber 176) untermauert. Kein anderer Autor hatte einen ähnlich schnellen Rechner.

Die dritte Weltmeisterschaft wurde 1980 in Linz, Österreich, ausgetragen. Dort traten zwei Wissenschaftler der Bell Laboratories mit einer selbstgebauten Schachmaschine an, die wesentlich kleiner – und hundertmal billiger – als der Cyber war. Dennoch konnte die Spezialkonstruktion, die auf den Namen Belle getauft wurde, schachspezifische Berechnungen weitaus schneller als der Superrechner abarbeiten. Es kam zu einem spannenden Wettkampf, bei dem Belle die gesamte Konkurrenz abhängte und dritter Computerschach-Weltmeister wurde.

Drei Jahre später trat eine Mannschaft aus Mississippi an, um mit dem Programm Cray Blitz die Ehre der Großrechner zu retten. Robert Hyatt, Al Gower und Harry Nelson hatten dem schnellsten Computer der Welt das Schachspielen beigebracht. Der Cray-I XMP kann in einer einzigen Sekunde nicht weniger als 210 000 000 Rechenoperationen durchführen – weit mehr als ein Mathematiker mit Papier und Bleistift in einem Menschenleben. Normalerweise bedient die Supermaschine tausende von Anwendern gleichzeitig, aber bei der vierten Computerschach-WM 1983 in New York stand ihre gesamte Rechengewalt im Dienste des Schachspiels. Das zahlte sich aus: Cray Blitz schlug Belle und gewann die Weltmeisterschaft.

Auch bei der WM in Köln war man gespannt auf den Zweikampf David gegen Goliath. Denn bei der letzten Nordamerikanischen Computerschach-Meisterschaft siegte nicht wie gewohnt der amtierende Weltmeister Cray Blitz, sondern ein neues Programm namens Hitech. Die meisten Computerschach-Freunde hatten noch nie von ihm gehört, aber bereits zum Zeitpunkt des Turniers war Hitech bei Turnierspielern in den Vereinigten Staaten bestens bekannt: In Wettkämpfen gegen Menschen hatte er hervorragende Erfolge erzielt.

Die neue Schachmaschine wurde an der renommierrten Carnegie-Mellon-Universität in Pittsburgh entwickelt und stellt in erster Linie eine sehr schnelle Schach-Hardware dar. Während eine Cray

mit ihren 210 Millionen Operationen es schafft, etwa 30 000 Schachstellungen pro Sekunde zu erzeugen und zu bewerten, bringt es Hitech mit seinen 64 speziell für das Schachspiel entwickelten VLSI-Chips auf atemberaubende 175 000 Stellungen in der Sekunde. Ein Sun-1.6-Mikrocomputer besorgt die Kommunikation mit der Maschine und versorgt sie mit kontextabhängiger Schachinformation. Da gerade hier ständig neue, bessere Bewertungsalgorithmen eingesetzt werden, gibt es Grund zu der Annahme, daß Hitech die erste Maschine sein könnte, die Großmeister-Stärke erreicht.

In letzter Zeit wurde eine andere Entwicklung deutlich: Immer mehr Schachprogrammierer steigen auf Mikrosysteme um. Das liegt vor allem daran, daß Prozessoren wie der 68000 von Motorola heute Leistungen erbringen, die vor wenigen Jahren den Großrechnern vorbehalten waren. Ein solcher Prozessor verrichtet beispielsweise in dem deutschen Schachcomputer Mephisto Amsterdam seine Arbeit. Dieses käuflich erhältliche Holzgerät (es kostet ca. 3000 Mark und wird in München von der Firma Hegener + Glaser gebaut) siegte 1985 bei der Mikrocomputer-Weltmeisterschaft in Amsterdam mit 22 Punkten aus 24 Partien – ein kaum vorstellbares Ergebnis! Für eine Plazierung unter den ersten vier schien der verbesserte Mephisto Cologne, der zudem mit superschneller 68020-Hardware lief, allemal gut.

Modernste Kommunikationstechnik

Bei früheren Weltmeisterschaften bedienten sich die meisten Programmierer noch der guten alten Telefonleitung: Sie sprachen mit einem Kollegen, der im Rechenzentrum saß und den eigentlichen Dialog mit dem Computer führte. Zeitverluste und Mißverständnisse waren unvermeidlich, und die hohen Kosten für die Standleitungen machten die Veranstaltung zu einer sehr kostspieligen Angelegenheit. Für die WM in Köln hatte dagegen ein führendes Unternehmen für elektronische Datenfernübertragung, die Deutsche Mailbox GmbH in Hamburg, es tatsächlich fertiggebracht, alle extern stationierten Großrechner über den weitaus preiswerteren Datex-P-Service der Bundespost mit dem Austragungssaal in Köln zu verbinden. Dabei wurden die anfallenden Daten in einem Knotenrechner gesammelt und über eine gemeinsame (virtuelle) Leitung nach USA, Kanada oder England geschickt. Berechnet wird bei Datex-P nur die aktuelle Übertragungszeit – mitunter Bruchteile eines Pfennigs pro Zug. Das alles geschah fast ohne Zeitverzögerung, so daß der Bediener eines Rechners in Neu-Mexiko fast das Gefühl hatte, sein Terminal sei mit einem Computer im Nebenzimmer verbunden.

Natürlich verfügte nicht jeder Rechner über einen Datex-P-kompatiblen Anschluß, und so mußte das Kommunikationsgenie der Deutschen Mailbox, Horst-Günther Lynsche, am Vorabend der WM so manch abenteuerliche Verbindung stricken. („Wenn Sie mit Ihrem System eine Verbindung von Minnesota über Alabama nach Alberta herstellen können, dann klinken wir uns einfach in Kanada in das Netz ein!") Auch eingefleischte Computerspezialisten waren von so viel Improvisationstalent beeindruckt, und am Ende des Turniers mußten sogar die Skeptiker gestehen: Es hat alles tadellos geklappt.

Mikros auf dem Vormarsch

Keine Titelaspirationen in Köln hatten dagegen die kleinen 8-bit-Programme, die in den „normalen" Schachcomputern ihre Arbeit verrichten. Die kleinsten unter ihnen – gerade taschenbuchgroß und 350 Gramm schwer – kosten ja auch wesentlich weniger als eine CPU-Minute auf der Cray. Sie sind für die Masse der Hobbyspieler ein schönes Weihnachtsgeschenk und ein geduldiger Schachpartner.

Ein solches 8-bit-Programm heißt Rebel und wurde von dem Holländer Ed Schröder für den 6502 entwickelt. Ursprünglich sollte er 1985 in einem Mephisto-Schachcomputer eingesetzt werden, aber die Hersteller waren noch nicht von seiner Spielstärke überzeugt. Schröder ging zurück an die Arbeit und bastelte ein halbes Jahr an dem Programm. In Köln erschien er mit der neuesten Version, die auf einer mit 11 MHz getakteten 6502 lief. Die Hardware

war nicht besonders stabil und mußte mit einem Handventilator gekühlt werden. Aber Rebel gewann eine Partie nach der anderen – und sorgte beinahe für die größte Sensation der Computerschachgeschichte.

Als die letzte Runde in Köln eingeläutet wurde, mußte der bis dahin souverän führende Hitech die erste Niederlage gegen Cray Blitz hinnehmen. Am Nebenbrett spielte Rebel gegen die Spezialmaschine Be-Be, die schon seit Jahren als Anwärter auf den Titel gilt. Aber ein Blick auf das Brett verriet: Rebel stand klar auf Gewinn! Die gesamte Halle begann zu rechnen. Würde Rebel die Partie für sich entscheiden, wäre er unter den Erstplazierten aufgrund der besten ,,Buchholzwertung" (ein bei solchen Turnieren eingesetztes System zur Bestimmung der Plazierung bei Punktgleichheit) tatsächlich Weltmeister. Für Rebel-Autor Ed Schröder wurde es eine herzzerreißende Partie, die schließlich doch von Be-Be gewonnen wurde (siehe Kasten auf dieser Seite).

Um ein Haar hätte es also einen Weltmeister aller Klassen gegeben, den man für einige hundert Mark ins Haus holt. Auch so konnte sich die Rebel-Mannschaft vor Anfragen kaum noch retten: Wann kommt das Programm auf den Markt, wieviel wird es kosten? Bekannt war allerdings nur, daß Hegener + Glaser das Programm noch in diesem Jahr unter der Bezeichnung MM-III als Zusatzmodul für ihre Mephisto-Schachcomputer auf den Markt bringt. Natürlich wird es nicht so schnell laufen wie in Köln, aber das dürfte die wenigsten Schachcomputer-Freunde stören: Auch mit 3 oder 5 MHz dürfte es zu den stärksten Mikrocomputer-Programmen der Welt gehören.

Den Titel des Weltmeisters behielt zum Schluß Cray Blitz, punktgleich mit Hitech, Be-Be und Phoenix, aber mit der besten Buchholzwertung. Der verwendete Rechner stand in den Forschungslabors von Cray Research in Mendota Hights, Minnesota, und besaß vier CPU. Das machte 420 000 000 Operationen pro Sekunde und ist gut für 100 000 Stellungsbewertungen in der Sekunde. So viel nackte Rechengewalt ist bisher noch nie für das Schachspiel eingesetzt worden.

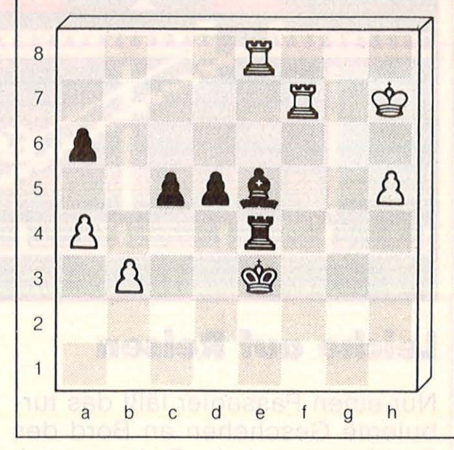
Frederic Friedel

Die Partie, mit der Rebel die Weltmeisterschaft verspielte

5. Runde: Rebel gegen Be-Be

1.d4 Sf6 2.c4 c5 3.d5 e6 4.Sc3 exd5 5.cxd5 d6 6.e4 g6 7.Lf4 a6 8.Sf3 Lg4 9.Le2 Db6 10.Dd2 Lg7 11.0-0 0-0 12.h3 Lxf3 13.Lxf3 Sbd7 14.Tad1 Tfe8 15.b3 Se5 16.Le2 Db4 17.Dc2 Te7 18.Lg3 Tae8 19.Tfe1 g5 20.Tf1 Kh8 21.Tc1 h5? Bisher hat Schwarz die Eröffnung recht gut behandelt, nach klassischem Muster, während Weiß nur planlos herumzog. Doch nun macht er einen schwerwiegenden Fehler (besser war 21 ... h6) und verliert damit beinahe die Partie.

22.f4 gxf4 23.Txf4 Sg6 24.Tf5 Sxe4 25.Sxe4 Dxe4 26.Dxe4 Txe4 27.Lxh5 Se7 28.Txf7 Ld4+ 29.Kh1 Sxd5 30.Txb7 Td8 31.Lf3 Te3 32.Lh4 Sf6 33.Tf7 Te6 34.Ld5 Sxd5 35.Lxd8 Sb4 36.a3 Kg8 37.Tcf1 Sc2 38.Tf8+ Kg7 39.a4 d5 40.h4 Se3 41.T1f7+ Kg6 42.Tc7 Sd1 43.Tg8+ Kf5 44.Tf7+ Ke4 45.g4 Kd3 46.h5 Te1+ 47.Kg2 Se3+ 48.Kg3 Le5+ 49.Kh4 Th1+ 50.Kg5 Tg1 51.Kg6 Txg4+ 52.Lg5 Tb4 53.Lxe3 Kxe3 54.Te8 Tg4+ 55.Kh7 Te4

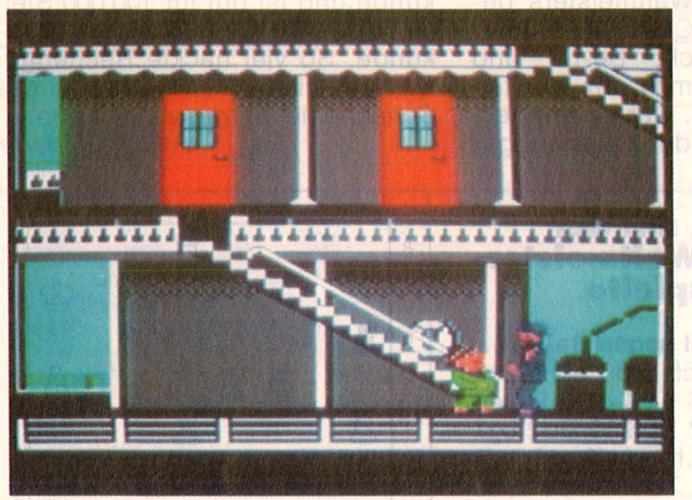


Die Stellung nach dem 55. Zug

56.Ta7?? Dieser Zug verliert Millionen! Statt mit 56.Tfe7 Kd4 57.Txe5 Txe5 58.Txe5 Kxe5 59.Kg6 d4 60.h6 die Partie für sich zu entscheiden (Weiß wandelt als erster mit Schach um), ging Rebel völlig unnötig auf Bauernjagd. Er verschenkte damit die Chance, Weltmeister zu werden – und das ganz große Geschäft zu machen.

56 ... d4 57.Txa6 d3 58.Tg6 d2 59.Tg1 Kf2 60.Teg8 Te1 61.T1g2+ Ke3 62.Txd2 Kxd2 63.Tc8 Ld4 64.Tb8 Te6 65.Tb7 Kc2 66.b4 c4 67.b5 c3 68.Td7 Kd3 69.b6 c2 70.b7 c1D 0:1

SPIEL DES MONATS



Was nun, Sir Foxworth? *Die Detektive, ratlos auf dem Raddampfer: Hinter einer der vielen Türen verbirgt sich der Killer und legt ein Ruhepäuschen ein – vor dem nächsten Mord.*

Leiche auf Reisen

Nur einen Passagier läßt das turbulente Geschehen an Bord des Raddampfers kalt: Er liegt mausetot in seinem Blute und in Kabine 4 der „Delta Princess". Und während sich das Schiff weiter den Mississippi hinunterschaufelt, New Orleans entgegen, beginnt die Suche nach dem Mörder. Sir Charles Foxworth und sein Detektivgehilfe Regis Phelps durchstöbern den alten Kahn von unten bis oben, vom Maschinenraum bis zur Brücke – und alle drei Decks dazwischen. Sehr zum Leidwesen von Besatzung und Passagieren, die anfangs weder für drängende Fragen noch für lästige Durchsuchungen viel Verständnis aufbringen. Immerhin, es finden sich Indizien – parfümierte Putzlappen, ein Pistolenfutteral, ein altes Ticket . . . und aus dem Gewirr der Aussagen kristallisieren sich nicht nur die persönlichen Eigenheiten der Reisenden und ihre zwischenmenschlichen Beziehungen immer deutlicher heraus, sondern auch ein handfester Tatverdacht. Sir Foxworth kann die Handschellen zuschnappen lassen (Falls ihn der Killer nicht schon vorher ins Jenseits befördert hat).

Mit dem Kriminal-Computerspiel „Murder on the Mississippi" landete Activision einen doppelten Volltreffer: Die sehr verschiedenen ausgeprägten Charaktere der Beteiligten werden aus ihren Dialogen und den Kommentaren präzise und originell gezeichnet, schon in der englischen Version überrascht ein überdurchschnittliches sprachliches Niveau (zwischen Oxford und Slumslang).

Aufbereitung und Zutaten stimmen ebenfalls: Das prächtig animierte Detektivgespann arbeitet sich durch ein üppig ausgestattetes Schiff mit 29 Räumen voller Überraschungen und Rätsel durch, mit dem Joystick gesteuert und einem gut sortierten Fragenkatalog ausgestattet. Texteingabe per Tastatur erübrigt sich, ebenso die Kenntnis des Englischen – eine deutsche Version ist in Sicht. Findigkeit und Kombinationsvermögen sind gefragt, weniger dagegen Geschicklichkeit oder Ausdauer: Das Jagdfieber stellt sich von selber ein.



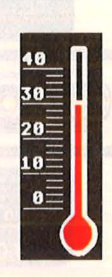
„Murder on the Mississippi"
Von Activision. Für Commodore 64/128. Diskette circa 60 Mark. Eine Delikatesse für Krimifans.



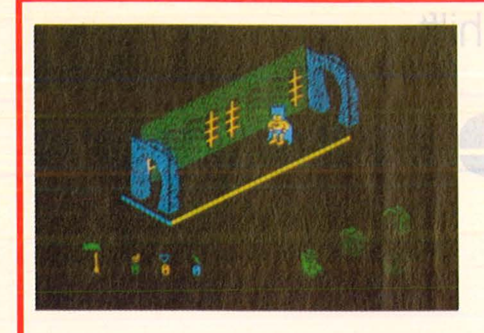
Bombenstimmung

Ein Drama mit drei Aufzügen und jeder Menge Meuchelmörder: In „Mission Elevator" klappert der heldenhafte Hausdetektiv alle 36 Stockwerke eines Hotels auf der Suche nach einer versteckten Bombe ab, immer in engem Kontakt mit der Unterwelt. Die schießt aus allen Rohren (und allen Lifts, Zimmern und Korridoren) auf den Sprengstoffschnüffler. Der soll in der bleihaltigen Atmosphäre auch noch diverse Gegenstände finden, das Personal bestechen, Pokern und die Bar besuchen.

Im Kugelhagel erweist sich der Detektiv als erstaunlich gelenkig: Nicht nur die Liftfahrten erinnern an „Impossible Mission", sondern auch die Animation des Hauptdarstellers. Das Ganze ist überdies recht witzig, detailfreudig und stimmungsvoll in Szene gesetzt und ergibt so eine aparte Mischung zwischen schneller Action und Such/Ratespiel. Sehr erfreulich: Der Hauptdarsteller darf sich knapp tausendmal pro Spiel umnieten lassen. Erforderlich sind Reaktionsvermögen und (geringfügige) Englischkenntnisse.



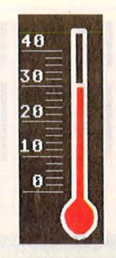
„Mission Elevator"
Von Rushware. Für Commodore 64 und Schneider CPC. Kassette circa 40 Mark und Diskette circa 50 bis 60 Mark. Die Mafia nimmt den Aufzug.



Flattermann im Anflug

Jeden Tag derselbe Ärger: Batman, der Fledermausförmige, sucht verzweifelt Stiefel, Gürtel, Triebwerk und Schulranzen, um heute endlich die Teile für sein „Batcraft" einsammeln zu können. Wie üblich sind die Monster von Joker & Riddler hinter ihm her und zehren an Lebens- und Nervenkraft. Die Suche nach den Fahrzeugteilen erstreckt sich über 151 Räume, die größtenteils mit Gerümpel und Schurken vollgestopft sind.

Dexter läßt grüßen: Die dreidimensionale Darstellung der Räumlichkeiten, das verblüffende Eigenleben mancher Gegenstände und die Animation der Darsteller gemahnen stark an die Games „Get Dexter" und „Fairligt", entfernt auch an „Spindizzy". Nur daß diesmal der Job des Helden noch komplizierter geraten ist. Dafür sorgen tückische Brücken und Fragen sowie die diversen Möglichkeiten, Energie oder Leben aufzutanken — oder zu verlieren. Durchschnittliche Geschicklichkeit und etwas Findigkeit genügen dennoch für diese haarsträubende Jagd über acht Level.



„Bat Man"
Von Ocean Software/Rushware. Für Schneider CPC. Kassette circa. 30 Mark.
Ein Held läuft heiß.



Barfuß durch den Friedhof

Erst gehen sie dem Rittersmann an die Wäsche, dann ans Leben: Die wüsten Geister, die sich aus den Gräbern des noch wüsteren Friedhofes erheben, legen nicht nur ein beachtliches Tempo an die Nacht, sie haben außerdem einen harten Schlag am (Astral-) Leib. Der Ritter steht nach dem Erstkontakt in der Unterwäsche da, nach dem zweiten in den Knochen. Er wehrt sich gegen den morbiden Strip unter anderem mit Feuerkugeln und Wurfschwertern — wenn er nicht gerade über Grabsteine oder Grüfte springt, Leitern erklettert oder Drachen niedermacht. Der Hindernislauf führt über vier Level, das Land der Elfen und letztlich in ein Schloß, wo sich das ganze geisterhafte Gesindel nochmal ein letztes Stelldichein gibt.

Ein geradezu klassisches Action- und Ballerspiel, das neben gruseliger Stimmungslage auch eine hübsche grafische Umsetzung mitbringt. Die musikalische Begleitung des Grusicals kann sich ebenfalls hören lassen. Durchaus einfach per Joystick zu steuern, verlangt das Spiel nach einiger Reaktionsschnelligkeit — immerhin gibt es nur sechs Leben pro Runde zu verlieren.



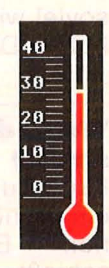
„Ghosts'n Goblins"
Von Elite. Für Schneider CPC und Commodore 64: Kassette circa 34 Mark, Diskette circa 53 Mark.
Schneller Grabsteinslalom



Liebestolle Meerjungfrau

Mit der ganzen Fülle ihres wahrlich gewaltigen Luxuskörpers wirft sich die Meerjungfrau Myrtle in die Fluten des Ozeans, auf der Jagd nach einem verzweifelt flüchtenden Taucher. Der findet kurzfristig Schutz unter dem Wrack eines versunkenen Dampfers, wo er traurig wartet — bis ihm der Sauerstoff ausgeht, oder die Nymphe sein feuchtes Versteck entdeckt. Eine schauerliche Alternative.

Die Meerjungfrau pflügt derweil durch die tückischen Tiefen, wo allerlei Gefahren auf sie lauern — in Form von Haien, Kraken und Seeungeheuern. Ihre schwindenden Energievorräte bekämpft sie mit dem Inhalt einiger Guinessflaschen, die am Meeresgrund herumliegen. Auch sonst gibt es allerlei interessante Objekte zu finden. Endlich mal ein Spiel, das sich nicht so tierisch ernst nimmt, eine echte Erholung zwischen Weltraumdramen und Karateprügeleien. Die originelle Handlung wurde in ebenso witzige Grafik umgesetzt. Dazu kommt eine wohlgeratene Animation und eine vorbildliche Bildschirmaufteilung: Der größte Teil der Fläche bleibt der Handlung reserviert, nicht den Accessoires.

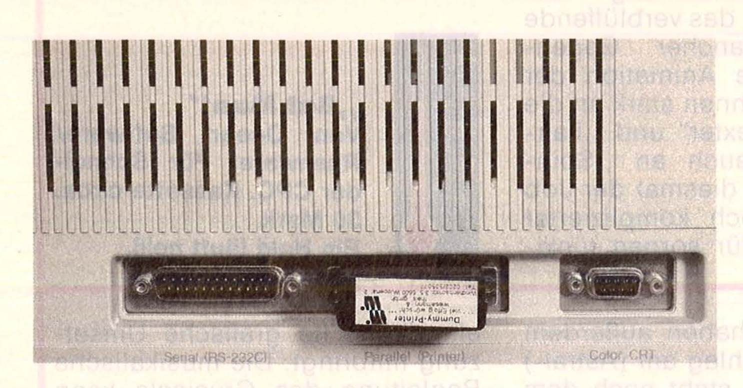


„Mermaid Madness"
Von Electric Dreams/Activision. Für 64, CPC und Spectrum. Kassette circa 40 Mark, Diskette circa 60 Mark.
Originelles Labyrinthspiel



Wo andere Tools versagen, hilft

Der letzte Hammer



Tarnen und Täuschen

Ein hitverdächtiges Computer-Zubehör hat die Firma Wiesemann entwickelt – den ,,Dummy-Printer". Das Ding kostet im Gegensatz zu einem Drucker, für den mindestens 500 Mark zu löhnen sind, nur zehn Mark. Dennoch leiste es Erstaunliches, meint der Hersteller: ,,Dieser kleine Stecker wird einfach auf den Druckerausgang eines IBM PC oder Kompatiblen gesteckt und simuliert einen angeschlossenen Drucker. Jetzt kann der Computer nach Herzenslust ausdrucken, ohne daß dafür unsere Wälder abgeholzt werden müssen."
Bravo! Der Simulant dürfte sich besonders in der Textverarbeitung positiv auswirken, wenn der Adressat nicht mehr mit dem geprinteten Unfug des Verfassers behelligt wird. Wir warten jetzt sehnlichst auf den Dummy-Monitor, die Dummy-Floppy und den Dummy-Hacker. Der Effekt: Garantiert nie mehr Ärger mit dem Computer.

Die Horror-Boxen

Schreckensmeldung aus den USA: In den Mailboxen machen sich Killerviren breit, die sich auf jeden harmlosen Computer stürzen, der mit der verseuchten Box Kontakt aufnimmt. Wie das US-Magazin ,,Omni" (und anschließend der ,,Spiegel") berichtete, verbreiten gemeine Freaks über Mailboxen infizierte Programme, die nach dem Abrufen sämtliche Datenbestände im Home-Computer sowie in der beteiligten Floppy gründlich ruinieren. Auf dem Monitor taucht kurz vor dem endgültigen Datenzerfall noch ein hämisches ,,Arf, arf! Gotcha!" auf – was soviel wie ,,Har, har" Wir haben Dich!" bedeutet.

Wenn der Bildschirm strahlt

Kaum zu glauben, was so ein Monitor alles auf seinen Betrachter abschießt. Zum Beispiel UV- und Röntgenstrahlen sowie Mikrostaubpartikel (20 000 VBE)! Brrrrrr. Dagegen hilft neuerdings die dezent getönte ,,Splend-Computer-Brille" in ,,Classic" (Gold) oder ,,Sport" (Schwarz) von der P. Nicola GmbH. Sie soll außerdem Ermüdung und Konzentrationsabbau mindern. Angesichts von Programmen wie Locoscript, Green Beret und Little Computer People fordern wir jedoch die garantiert undurchsichtige Computer-Brille. Das Design und das Gestell sind uns schnurz – die Hauptsache, gnädiges Schwarz verhüllt das Geschehen auf dem Monitor.

Gegendarstellung

Tausende von Atari-Besitzern erregten sich bis an den Rand des Herzinfarkts: Hatten wir doch im vorletzten Hammer (HC Juli 86) gefrevelt, daß der 800XL neben dem Amiga popelig wirke. Wir behaupten also hiermit keineswegs das Gegenteil, sondern bezeichnen feierlich alle IBMs, Commodores, Sinclairs, Vaxens, Schneiders sowie die Kisten von Pentagon, CIA, Verfassungsschutz und Vatikan als popelige Computer. Hoffentlich sind jetzt alle gleichmäßig beleidigt – oder sollten wir noch jemand vergessen haben?

Geburtsanzeige

Im Peek- und Poke-Dschungel des Commodore 64 existiert auch eine Speicherstelle für Patrioten: Wer wissen will, wo sein C64 gebaut wurde, braucht nur die Adresse 678 abzufragen: Taucht eine 1 auf, so läßt dies auf ,,Made in Germany" schließen, eine 0 dagegen deutet auf die Herkunft aus den USA hin (Die Betriebssystem-Programmierer schienen demnach an latentem Antiamerikanismus zu leiden). Daß Taiwan-Importe den Befehl PRINT PEEK (678) mit Absturz quittieren, ist ein unbestätigtes Gerücht.

Joachim Graf

Electronic Killer

Genau das haben wir uns schon immer gedacht, und jetzt stand es sogar in der Abendzeitung (München): ,,Computer ließ Rentnerin ‚sterben'." Die Dinger sind also gar nicht so harmlos, wie sie immer tun. Aber schon in der Unterzeile wird klar, daß unser elektronischer Hausfreund wie üblich halbe Arbeit geleistet hat: ,,Frau, (75), kämpft: ‚Ich lebe'". Der Amateurmörder pfuscht in der Bundesversicherung für Angestellte und hatte die höchst lebendige Leiche versehentlich (oder absichtlich?) auf die Liste verblichener Rentenempfänger gesetzt. Requiescat in pace.



Impressum

Redaktionsdirektor: Richard Kerler

Chefredakteur: Wolfgang Taschner (verantwortlich für den Inhalt)

Art Direction: Hans Kuh

Chef vom Dienst: Marianne Weißbach

Redaktion: Hans Schmidt (stellv. Chefredakteur), Horst Brand, Reinhardt Hess, Dieter Winkler

Bildredaktion: Konstantin Kern, Iris Klaus

Redaktionsassistentin: Isabella Feig

Grafische Gestaltung: Antonia Graschberger, Gabi Klotz

Illustration: Arnold Metzinger

Fotografie: Ezio Geneletti, Franz-K. Hummel, Studio eins

Autoren dieser Ausgabe: Dr. Siegfried Bagdonat, Wolfgang Börner, Joachim Graf, Alfred Görgens, Frederic Friedel, Thomas Geise, Wolfgang Heider, Reiner Kunz

Redaktion: Vogel-Verlag KG Würzburg, Redaktion HC, Schillerstr. 23a, 8000 München 2, Telefon (0 89) 51 49 30, Teletex 89 71 90, Telex 17-897 190, Telefax (0 89) 53 50 00

Verlag: Vogel-Verlag KG, Postfach 67 40, D-8700 Würzburg 1, Tel. (09 31) 41 02-1, Telex 6 8 883, Telefax (09 31) 41 02-5 29, Telegramme: HC Würzburg

Anzeigenleiter: Harald Kempf, Würzburg (verantwortlich für Anzeigen)
Anzeigenservice: HC, Postfach 67 40, 8700 Würzburg, Tel. (09 31) 41 02-1, Telex 6 8 883.
Michael Belgrad, Durchwahl 41 02-4 33.
USA: Jay Eisenberg, 6855 Santa Monica Blvd. Suite 202, Los Angeles, CA 90038, Tel. (2 13) 4 67-22 66, TWX 91032–13134
Anzeigenpreise: z.Z. gültig Anzeigenpreisliste Nr. 2 v. 1.1.85

Vertriebsleiter: Axel Herbschleb, Würzburg
Vertrieb Handelsauflage: Inland (Groß-, Einzel- u. Bahnhofsbuchhandel): Vereinigte Motor-Verlage GmbH & Co. KG, Leuschnerstr. 1, 7000 Stuttgart 1, Tel. (07 11) 20 43-1, Telex 7 22 036. Ausland: Deutscher Pressevertrieb Buch-Hansa GmbH, Wendenstr. 27–29, 2000 Hamburg 1, Tel. (0 40) 2 37-11-1, Telex 2 162 401
Vertriebsvertretungen: Österreich: Erb Verlag GmbH & Co. KG., Amerlingstr. 1, A-1061 Wien 6, Tel. (02 22) 56 62 09, Schweiz: Thali AG, CH-6285 Hitzkirch, Tel. (0 41) 85 28 28
Erscheinungsweise: monatlich.

Bezugspreis: Jahresabonnement Inland 55,– DM (51,40 DM + 3,60 DM Umsatzsteuer), Ausland: in Österreich 470 öS, in der Schweiz 59,– sfr., sonstige Länder 64,– DM. Abonnementspreis inkl. Versandkosten Einzelheft Inland 5,– DM (4,67 DM + 0,33 DM Umsatzsteuer), Ausland: 5,50 DM, Einzelpreis + Versandkosten.
Bezugsmöglichkeiten: Bestellungen nehmen der Verlag, die o.a. Generalvertretungen, jedes Postamt und alle Buchhandlungen im In- und Ausland entgegen. Abbestellungen sind nach Ablauf der Mindestbezugszeit bei einer Kündigungsfrist von 2 Monaten jeweils zum Quartalsende möglich. Sollte die Zeitschrift aus Gründen, die nicht vom Verlag zu vertreten sind, nicht geliefert werden können, besteht kein Anspruch auf Nachlieferung oder Erstattung vorausbezahlter Bezugsgelder.
Bankverbindungen Vogel-Verlag: Dresdner Bank AG, Würzburg (BLZ 790 800 52) 314 889 000; Bayerische Vereinsbank AG, Würzburg (BLZ 790 200 76), 2 506 173; Kreissparkasse, Würzburg (BLZ 790 501 30) 17 400; Postscheckkonto Nürnberg (BLZ 760 100 85) 99 91–8 53
Ausland: Postscheckkonto Zürich 80-47 064; Postscheckkonto Niederlande 2 66 23 95; Banque Veuve Morin-Pons, Paris, 1 55 41 03 14

Satz, Litho, Druck, Verarbeitung und Versand: Vogel-Druck Würzburg



VOGEL Würzburg

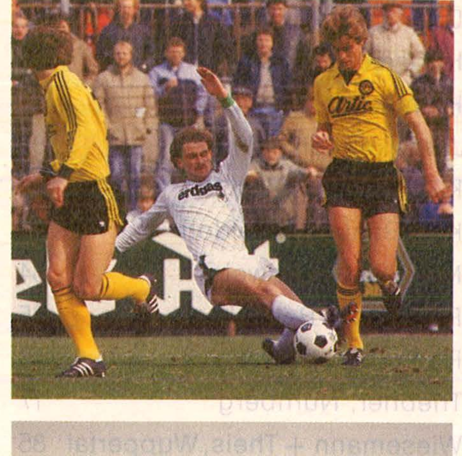
Inserentenverzeichnis

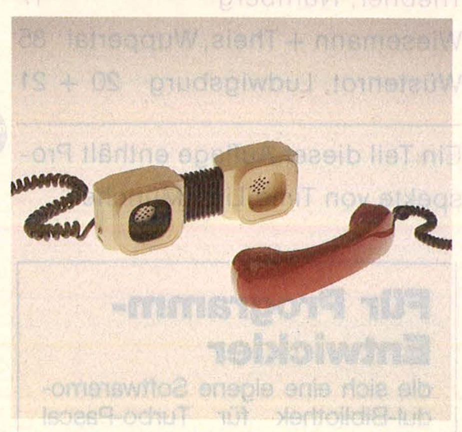
Ein Teil dieser Auflage enthält Prospekte von Time Life, München.

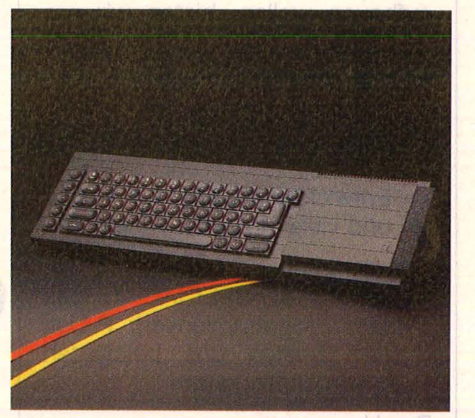


VORSCHAU

Das nächste Heft erhalten Sie ab 15. September 1986 bei Ihrem Zeitschriftenhändler







Die Bundesliga total erfaßt: Ein Superlisting für Commodore 64 bringt jederzeit Ordnung und Überblick in das (hoffentlich) dramatische Geschehen in den Fußball-Stadien. Das patente Tabellenprogramm wird natürlich auch mit allen anderen Liga-Sportarten fertig.

Datenfernübertragung, mehr als ein faszinierendes Hobby: Was der persönliche Kontakt zu Datenbanken und Mailboxen bringt, wie der ideale DFÜ-Platz aussieht, welche Terminal-Programme und Akustikkoppler sich am besten bewähren – HC liefert das komplette Know-how für Datenreisende.

Sinclair QL, das verkannte Genie: Wir stellen zusammen, was für den meist unterschätzten Rechner derzeit an Software und Zubehör angeboten wird. Dazu eine informative Beschreibung sämtlicher Schnittstellen des „Quantensprungs".

Außerdem lesen Sie:

CAD und CPC vertragen sich prächtig. Den Beweis liefert das Programm CAD-Easy, das auf allen Schneider CPC läuft.

Im Pascal-Teil müssen diesmal alle einschlägigen Compiler zeigen, was sie können.

Forth für Individualisten: Die Programmiersprache läßt sich mit eigenen Befehlen beliebig erweitern.

Mit Desktop-Programmen kommt Luxus am Computer auf. HC beschreibt, was die nützlichen Helfer im Hintergrund leisten.

Drei neue Drucker stellen wir vor: Von Atari, Riteman und Robotron.

Ein Sound-Sampler entlockt dem Atari ST die ungewöhnlichsten Töne. Wir testen den Tonkünstler.



VOGEL Computerbücher

Senftleben, Dietrich
Start mit Apple-Logo für II, IIe und IIc
Das kleine Logo-Einmaleins
Grafik · Text · Musik
222 Seiten, 70 Abbildungen, 35,— DM, 1985
ISBN 3-8023-**0832**-8

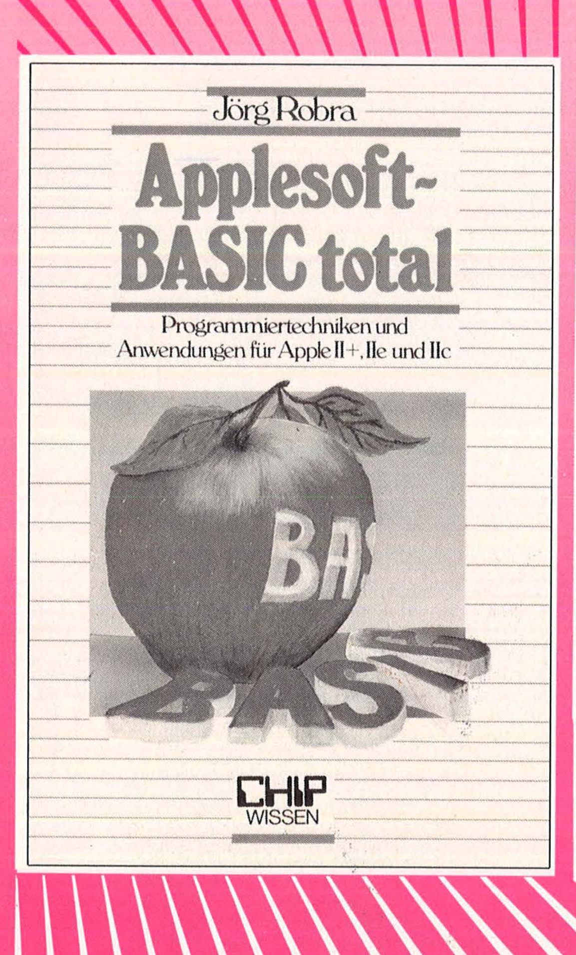
Willkommen bei Logo und seinen beiden Versionen Apple-Logo und Apple-Logo II, bekannt als benutzerfreundliche Computersprache für Ausbildung, Freizeit und Beruf. Mittels Schildkrötengrafik wird das kleine Logo-Einmaleins in 12 Lektionen beschrieben. Das Buch verlangt praktische Mitarbeit. Es hat seinen Platz neben dem Computer und gibt Hilfen und Anregungen für eigenes Forschen.

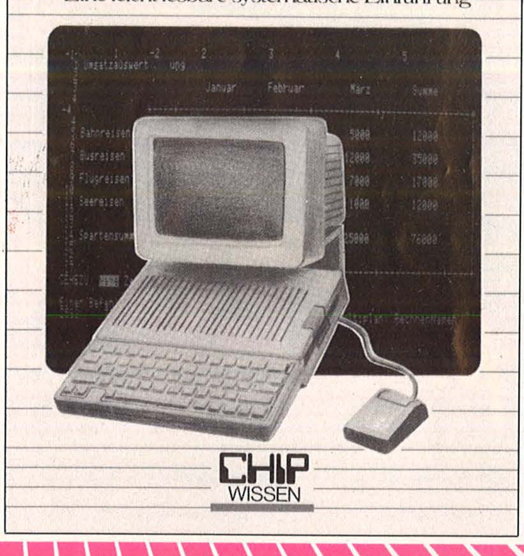
Teiwes, Eike
Programmentwicklung in UCSD-Pascal
Beispiele · Aufgaben · Anregungen
344 S., zahlr. Abb., 28,— DM
ISBN 3-8023-0760-7

Aus dem Inhalt:
— Das ist Pascal
— Rechner und Betriebssystem
— Programmentwicklung
— Datentypen
— Grafik
— Grafik im Textmodus
— Dateneingabe und -ausgabe u.v.m.

Baumann, Rüdeger
Spiel, Idee und Strategie programmiert in Pascal
326 S., zahlr. Abb., 35,— DM
ISBN 3-8023-0732-1

Aus dem Inhalt:
— Denkspiele mit Zahlen
— Suchen und Raten
— Solospiele
— Zweipersonenspiele
— Spielstruktur
— Spielanalyse
— Suchalgorithmen u.v.m.





VOGEL-BUCHVERLAG WÜRZBURG

Vogel-Computerbücher — mehr wissen, mehr leisten

Sie erhalten bei Ihrem Buch- und Computerfachhändler kostenlos das neue Verzeichnis **„VOGEL-Computerbücher 1986"** mit ca. 120 aktuellen Titeln unserer Reihen **CHIP WISSEN** und **HC — Mein Home-Computer.**

Robra, Jörg
Applesoft-BASIC total
Programmiertechniken und Anwendungen für Apple II+, IIe und IIc
340 Seiten, zahlr. Abbildungen

Das Buch zeigt, wie aus einem Problem — zunächst computerunabhängig — über Analyse, Zieldefinition, Benutzerhandbuch, Modularisierung und strukturierte Programmierung ein sauber aufgebautes und benutzerfreundliches Programm entsteht.

ISBN 3-8023-0872-8 45,— DM

Kretschmer, Bernd
Multiplan-Training auf Apple IIe und Apple IIc
Eine systematische Einführung
260 Seiten, zahlr. Abbildungen

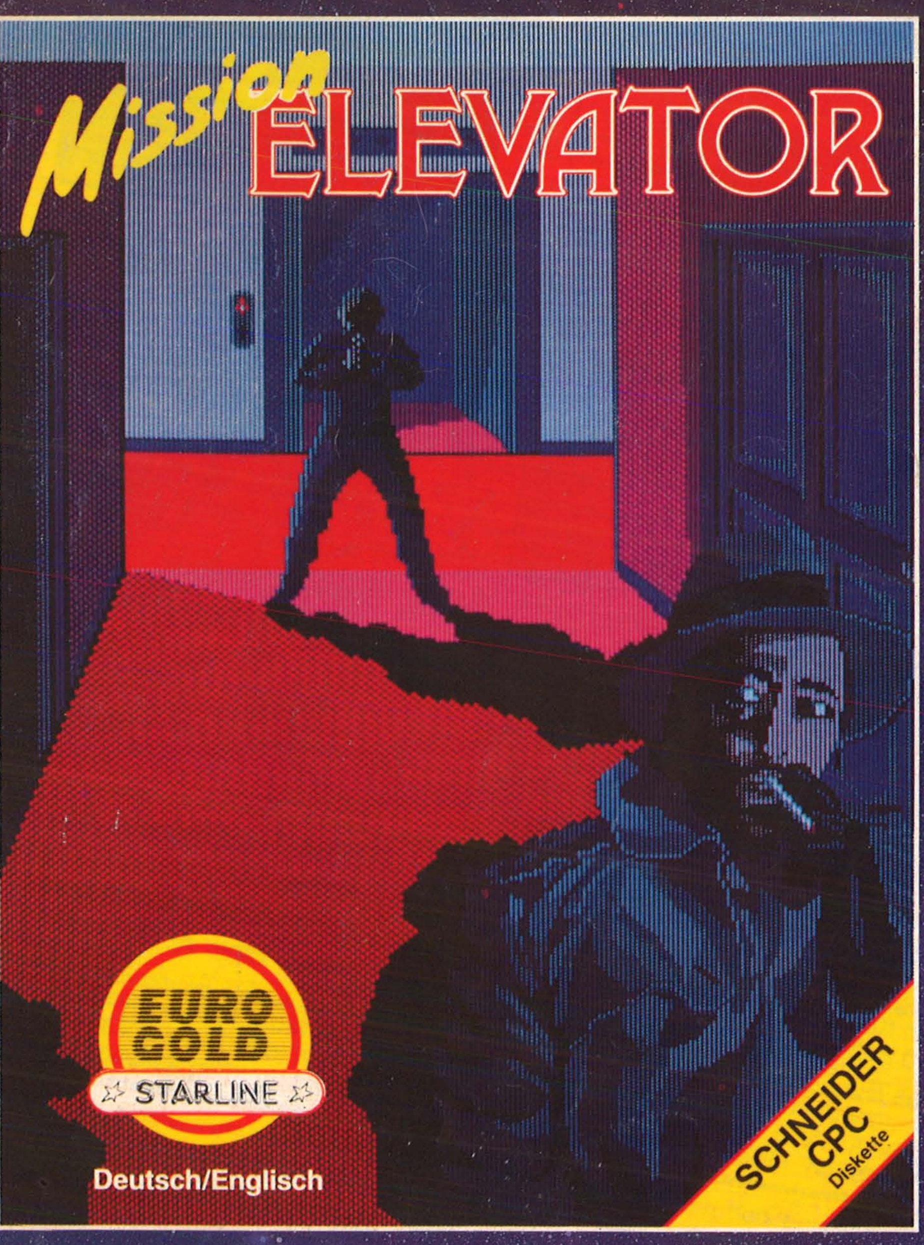
Alle Nutzer des Apple II werden hier ein Arbeitsmittel finden, das alles Wesentliche über das Tabellenkalkulationsprogramm Multiplan aussagt. Hier werden im didaktischen Ansatz Lösungsmöglichkeiten schrittweise anhand von vielen Beispielen aufgezeigt.

ISBN 3-8023-0847-6 38,— DM









"...sehr gut gelungenes Actionspiel mit Abenteuer-Touch und sehr hohem Spielwitz."
(HAPPY COMPUTER 07/86)

Mission ELEVATOR ist lieferbar für:
COMMODORE und
SCHNEIDER
(Kassette, Diskette)

MIT DEUTSCHER SPIELANLEITUNG!

MICROPOOL Produkte erhalten Sie in den Fachabteilungen von Horten und Quelle International sowie in gutsortierten Computershops.

Mission ELEVATOR bestellen Sie bei Quelle unter der Kat.-Nr.
CPC-Kass 560·632-2
C64-Kass 560·813-8
C64-Disk 560·823-7

Vertrieb:
RUSH WARE
Online with the trend.

Mitvertrieb:
Microhändler GmbH